A NORMALIZATION ALGORITHM FOR TRUNCATED P-ADIC ARITHMETIC

A. COLAGROSSI* - A. MIOLA**

* ISTITUTO SUPERIORE DI SANITA' - Viale Regina Elena 299, 00161 Roma, Italy
** DIPARTIMENTO DI INFORMATICA E SISTEMISTICA
UNIVERSITA' DI ROMA "LA SAPIENZA", Via Buonarroti 12, 00185 Roma, Italy

ABSTRACT

This paper presents a new algorithmic approach to cope with the problems related to the generation and the manipulation of the pseudo-Hensel-codes in the p-adic arithmetic.
After reviewing some classical properties and the results of the Hensel code arithmetic, a new algorithm to manipulate pseudo-Hensel-codes is presented, discussed and compared with two existing methods. The lower cost of the proposed new algorithm will result from the comparison.

1. INTRODUCTION

The problem of exact rational number arithmetic has received a wide attention from many years. Several successful approaches have been followed with relevant results [GRE80, H&H79, HOR77, KOM81, K&M81, MAT75].
One of the most significant techniques that has been applied is based on the p-adic construction proposed by Hensel [BAC64, KOB77, MAH73] and on the consequent use of the truncated representation known as Hensel-code arithmetic [GRE80, G&K84, MIO83, MIO84].
In this arithmetic each rational number is represented by its p-adic form, with respect to a given prime p, using only a fixed number r of its p-adic digits. The set of Hensel-codes, for given p and r, is referred as H(p,r).
Any problem involving rational number computations can be approached by using this Hensel-code arithmetic, according to the following classical schema:

a) convert the rationals, which represent the input in the given problem,into the corresponding Hensel-codes (i.e. go from Q to H(p,r));

b) perform the requested computations using Hensel codes arithmetic (i.e. operate in H(p,r));

c) convert back the result from the Hensel-code representation to the usual rational representation (i.e. go back from H(p,r) to Q).

Since H(p,r) is a finite set, whereas Q is an infinite one, some obvious problems could rise during the execution of step (c). However it has been shown that there exists a biunivocal corrispondence between a proper subset of H(p,r) and a subset of Q, namely the set of the Farey's fractions $F_{N(r)}$, the fractions which have their numerator and denominator limited by an integer N(r) depending on p and r. According to this result when the inputs to a computation belong to $F_{N(r)}$, if the image of the result in H(p,r) exists in $F_{N(r)}$, then such image is the exact result over the rationals [G&K84].
A problem still open is related to the manipulation of the so called pseudo-Hensel-codes. These codes are those which, being generated during a computation in H(p,r), for instance during an addition of Hensel-codes, suffer from a lack of normalization, (i.e. having leading zeroes in their sequence of p-adic digits). The possible ways to attach this problem are either to normalize to non-zeroes leading digits, by shifting the given code and proceeding with a smaller number of digits (i.e. the computation proceeds in H(p,s) with s < r), or to restore non-zeroes leading digits by shifting the given code and filling the code up to the r-th position by new significant digits.
The latter of these approaches, followed by Gregory and Krishnamurthy [G&K84], presents some errors and certainly fails in effective calculations, as shown in [DIT85].
The former approach, presented by Dittenberger in his thesis [DIT85], gives a complete solution to the problem, but it generates high computational complexity in those cases when the number of non-zeroes digits tends to reduce to zero and the entire computation is restarted from the

very beginning in $H(p,2r)$, (i.e. with a number of digits double than r).
A new approach is proposed in this paper and it will be shown to be preferable to the previous ones in many practical real situations.

2. APPROXIMATE P-ADIC ARITHMETIC: OPERATING WITH HENSEL-CODES.

In the Hensel-code arithmetic, each rational number c/d, with $\gcd(c,d)=1$, is expressed by its Hensel-code [GRE80], namely the triple:

(1) (α,p,r)

where:

$p \in N$ is a prime, $r \in N$, $r>0$,
$\alpha = (mant,exp)$,
with:
$mant = (.a_0 a_1 \ldots\ldots a_{r-1})$

$a_i \in Z_p$, $a_0 \neq 0$;
$exp \in Z$, defined as:

- if $\gcd(c,p) = \gcd(d,p) = 1$ then:

$$mant = (c(d^{-1} \bmod p)) \bmod p^r$$
$$exp = 0;$$

- if $\gcd(c,p) \neq 1$ or $\gcd(d,p) \neq 1$ then:

$$c/d = (c'/d')\, p^e ,$$

$$mant = (c'(d'^{-1} \bmod p)) \bmod p^r$$
$$exp = e.$$

Then the Hensel-codes are actually normalized Hensel-codes.The set $H(p,r)$ actually refers to the set of normalized Hensel-codes.
Computations with rational numbers can be performed applying the method proposed by Gregory and Krishnamurty [G&K84] with the following sequence of steps:

a) map the rationals from Q to $H(p,r)$ (direct mapping);

b) perform computations in $H(p,r)$;

c) map the results from $H(p,r)$ to Q (inverse mapping).

It is relevant to note the fact that the direct mapping always produces an unique image from Q to $H(p,r)$, while the inverse mapping could produce more than an image from $H(p,r)$ to Q, this being essentially due to the different structures of Q and of $H(p,r)$.
However, it has been shown that considering the set of Farey's fractions $F_{N(r)}$, which consists of all the rationals c/d with:

$$c < N(r), \quad d < N(r)$$

where

$$N(r) < \left\lfloor \sqrt{(p^r - 1)/2} \right\rfloor$$

there exists a biunivocal corrispondence between $F_{N(r)}$ and a proper subset of $H(p,r)$ [G&K84].
Under the hypothesis that the final result of the calculation belongs to $F_{N(r)}$, the Hensel-code arithmetic guarantees the exactness of the results even if during the calculation some overflow occurs (i.e. when intermediate elements belong to $F_{N(s)}$ with s>r are generated) [GRE80].

Because of that, when the inputs to a computation belong to $F_{N(r)}$,if the images of the results obtained in $H(p,r)$ exist in $F_{N(r)}$,then such images <u>are the exact</u> results over the rationals.

3. OPERATING WITH PSEUDO-HENSEL-CODES.

Let us now discuss the problem of the generation and the manipulation of the so called pseudo-Hensel-codes.
When adding two elements of $H(p,r)$ the result could happen to be in the following form:

$$\mu = (.0\ldots0\, c_k \ldots c_{r-1} ,exp).$$

This code is called pseudo-Hensel-code and, because of its lack of normalization, it should be normalized before proceeding with further computations.
The problem related to the treatement of pseudo Hensel-codes has already received two solutions by Gregory and Krishnamurty [G&K84] and by Dittenberger [DIT85].
Given the problem of executing a set of arithmetic operations over a set $(s_1,\ldots\ldots,s_n)$ of rationals numbers, let us now describe, more formally, the

algorithms proposed by Gregory and Krishnamurty (Algorithm GK) and by Dittenberger (Algorithm D).
Let Dir(t,r), for $t \epsilon Q$, and Inv(α,r), for $\alpha \epsilon H(p,r)$, denote respectively the direct and the inverse mappings. The Algorithm GK operates in the following way: delete all of the k leading zeroes of the given pseudo-Hensel-code obtaining a Hensel-code of size r-k, apply the inverse mapping and then the direct mapping, obtaining a Hensel code of size r.

Algorithm GK:

```
1. dir($s_i$,r) ---> $\alpha_i$ , for i=1,...,n;
2. perform a computation between Hensel-codes;
3. if during the computation in step 2 a pseudo Hensel-code
   $\mu$=(.0...0 $c_k$ ...$c_{r-1}$ ,exp), k>0,
   has been generated, then perform step 4, else perform step 5;
4. (normalization step)
   (.$c_k$ ....$c_{r-1}$ ,exp+k) ---> $\delta$,
   Inv($\delta$,r) ---> u,
   Dir(u,r) ---> $\mu$;
5. if $\mu$ is the final result then stop,
   else continue from step 2.
```

The algorithm D simply consists of deleting all the leading zeroes of the given pseudo-Hensel-code obtaining a Hensel-code of size r-k. When the size of the Hensel code becomes no longer significative, the computation needs to restart from the very beginning with Hensel codes of size 2r.

Algorithm D:

```
1. Dir($s_i$,r)---> $\alpha_i$ , for i=1,...,n;
2. perform a computation between Hensel codes;
3. if during the computation in step 2 a pseudo Hensel code
   $\mu$=(0...0$c_k$ ...$c_{r-1}$ ,exp), k>0,
   has been generated,then perform step 4,
   else perform step 5;
4. if r-k = 0 then perform step 6,
   else let
   (.$c_k$ ....$c_{r-1}$ ,exp+k) ---> $\mu$ ;
5. if $\mu$ is the final result then stop,
   else continue from step 2;
6. 2r ---> r,
   continue from step 1.
```

It is important to note how the algorithm GK sometimes fails, as shown by Dittenberger [DIT85]. An example of such a failure, proposed by Dittenberger, is the following.
Let p=5, r=4 be the fixed parameters and let a = 13/15, b = 13/10 be the inputs to perform a+b (=13/6).

Because:

```
        (13/15) ---> (.1413,-1) +
        (13/10) ---> (.4322,-1) =
                     ------------
                     (.0340,-1)
```

the result is a pseudo-Hensel-code (i.e. a_0=0).
The Algorithm GK at the step 4 furnishes the following values:

```
(.340,0) ---> δ
Inv(δ,4) ---> 3/11
Dir(3/11,4) ---> (.3403,0).
```

This result is clearly wrong because the correct result is 13/6 which corrisponds to (.3404,0) in H(5,4).
On the same case study the algorithm D furnishes a correct answer, even if this is correct only in H(5,3). In fact at the step 4, because k≠r (in fact k=1 and r=4), algorithm D sets (.340,0)--->μ then it continues the computation from step 2.
Let us underline that because of the shift in the step 4, algorithm D causes a loss of precision going from H(p,r) to H(p,r-k) everytime a pseudo-Hensel-code occurs. Consequently, algorithm D needs to restart the entire process, as described in step 6, in the following two cases:

a) when a pseudo-Hensel-code is generated with a number of significative digits less than a previously fixed value (in the proposed version of the algorithm D such a value is specified to be equal to zero, as in the step 4);

b) when a result in H(p,r-k) is obtained which has not an image in $F_{N(r-k)}$.

4. A NEW ALGORITHM TO NORMALIZE PSEUDO-HENSEL-CODES.

In order to avoid the loss of precision induced by the algorithm D, let us present a new algorithmic solution to the posed problem. The proposed Algorithm CM is based on the consideration that a pseudo-Hensel-code is generated by a sum. The first step consists in mapping both the operands of the sum over $F_{N(r)}$ and then over H(p,r+k). The second step consists in recomputing the sum over H(p,r+k). Finally, deleting k leading zeroes we have the right Hensel code over H(p,r).

Algorithm CM:

```
1. $\mathrm{Dir}(s_i,r) \text{--->} \alpha_i$, for i=1,...,n;
2. perform a computation between Hensel codes;
3. if during the computation in step 2 a pseudo Hensel code
   $\mu = (0...0\ c_k ... c_{r-1},exp)$, k>0,
   has been generated, then perform step 4, else perform step 7;
4. (note that $\mu=\alpha+\beta$, where $\alpha,\beta \in H(p,r)$)
   $\mathrm{Inv}(\alpha,r) \text{--->} u_\alpha$,
   $\mathrm{Inv}(\beta,r) \text{--->} u_\beta$,
5. (test on the existence of the inverses)
   if $u_\alpha$ and $u_\beta$ exist in $F_{N(r)}$,
   then perform step 6,
   else stop;
6. $\mathrm{Dir}(u_\alpha,r+k) \text{--->} \sigma$,
   $\mathrm{Dir}(u_\beta,r+k) \text{--->} \tau$,
$\sigma+\tau \text{--->} (.0...0\ c_k ... c_{r-1}\ \mu_r ... \mu_{r+k-1},exp)$
   $(.c_k ... c_{r-1}\ \mu_r ... \mu_{r+k-1},exp+k) \text{--->} \mu$;
7. if $\mu$ is the final result then stop,
   else continue from step 2.
```

In order to show how the algorithm CM works, let us apply it to the Dittenberger's example:
from p=5 and r=4 we have N(4)=17,and the Algorithm proceed as following:

1. $13/15 \text{--->} \alpha = (.1413,-1)$
   $13/10 \text{--->} \beta = (.4322,-1)$

2. $\alpha+\beta \text{--->} (.0340,-1)$

3. a pseudo-Hensel-code has been generated

4. $\mathrm{Inv}(\alpha,r) \text{--->} 13/15$
   $\mathrm{Inv}(\beta,r) \text{--->} 13/10$

5. the inverses exist in F_{17};

6. $\mathrm{Dir}(13/15,r+1) \text{--->} (.14131,-1)$
   $\mathrm{Dir}(13/10,r+1) \text{--->} (.43222,-1)$

$(.0...0\ c_k ... c_{r-1}\ \mu_r ... \mu_{r+k-1},exp) \text{---->}$
$\text{---->} (.03404,-1)$

$\mu \text{--->} (.3404,0)$

7. stop.

5. REMARKS AND COMPARATIVE ANALYSIS.

Let us compare now the Dittenberger's algorithm D with the new CM algorithm. Such an analysis will also define what situations provide the best performance for each algorithm.

Hypotheses.

Given a problem with n values as inputs, let us assume for the parameters involved in the analysis the following values:

n:= size of the input;

D(r)=I(r):= cost of Dir(u,r) and Inv(α,r); the cost of both algorithms is assumed to be the same, since they are both essentially based on the extended euclidean algorithm [MIO84];

M(r):= cost of an arithmetical operation between two elements of H(p,r);

m:= number of pseudo Hensel codes generated during the entire computation;

q:= number of operations performed between the generation of two consecutive pseudo Hensel codes.

Furthmore it is assumed that each pseudo Hensel code generated has only one leading digit equal to zero.

Theorem.

Under these hypotheses, if C(A) indicates the cost of an algorithm A, then:

1) when m < r and the result belongs to $F_{N(r-m)}$, it is C(D) < C(CM);

2) when m < r and the result belongs to $F_{N(r)}$ but it doesn't belong to $F_{N(r-m)}$, it is C(CM) < C(D) for every value of n if q > 2logr;

3) when m > r, it is C(CM) < C(D) in the same case as the point 2).

Proof.

Because both the algorithms D and CM perform the step 1, its cost won't be considered for the comparison. Furthermore the cost of shifting and testing operations won't be considered too.

Case 1), namely m < r and the result belonging to $F_{N(r-m)}$.
The algorithm D performs q times the step 2 for each of the m pseudo Hensel codes

generated, then the cost of D is:

$$mqM(r)$$

The algorithm CM perform q times the step 2 then perform the steps 4,5,6 and repeats this loop for m times; then the cost of CM is:

$$m(qM(r) + 2I(r) + 2D(r) + M(1)).$$

The assertion follows immediatelly.
Case 2), namely $m < r$ and the result belonging to $F_{N(r)}$ but not to $F_{N(r-m)}$.
The cost of the algorithm D increases with respect to the case 1) because it must restart the computation from the beginning with mantissas of size 2r (step 6).
Then the cost of D is:

$$mqM(r) + nD(2r) + mqM(2r)$$

By the assumed hypotheses, the cost of CM is:

$$mqM(r) + 4mD(r) + mM(1)$$

Being in this case the cost of the Extended Euclidean Algorithm rlogr, CM will have a lower cost than D when:

$$mqM(r) + 4mrlogr + mM(1) <$$
$$< mqM(r) + n(2rlog2r) + mqM(2r)$$

If we assume M(r) be at most r, we have:

$$(2) \quad n > \frac{m\ (\ 4rlogr - 2qr + 1)}{2rlogr}$$

For m = r it is:

$$n > \frac{r}{log2r}\ (\ 2logr - q\ ) + \frac{1}{2log2r}$$

i.e. $C(CM) < C(D)$ when $q > 2logr$.

Because for m = r the right member of (2) has an upper limit, the assertion follows.

Case 3. In this case the algorithm D must restart the computation with mantissas of 2r as length, then the same considerations as in 2) can be developed.

It can be remarked that the algorithm CM at the step 6 doesn't need to apply the direct mapping because it is sufficient to compute only k more digits for σ and τ.
It has to be noted, however, that the algorithm CM needs to perform inverse mappings for each pseudo Hensel code generated, and it should be guaranted the feasibility of such inverse mappings (see step 5 of algorithm CM).
However, one can obviously think to a sort of mixed approach: try to apply the CM algorithm and in case of stop at the step 5, apply the algorithm D.

REFERENCES

[BAC64] G.Bachman: Introduction to P-Adic Numbers and Valuation Theory. Academic Press, New York, 1964.

[DIT85] K.Dittenberger: An Efficient Method for Exact Numerical Computation, Diploma Thesis University of Linz 1985.

[GRE80] R. T. Gregory: Error - Free Computation. Robert E.Krieger Pub. Co.,Huntington, N.Y., 1980.

[G&K84] R.T.Gregory, E.V. Krishnamurthy: Methods and Applications of Error-Free Computation, Springer-Verlag 1984.

[H&H79] E.C.R.Hehner, R.N.S.Horspool: A New Representation of the Rational Numbers for Fast Easy Arithmetic. SIAM Journal on Computing, Vol. 8, pp. 124-134, May 1979.

[HOR77] B.K.P.Horn: Rational Arithmetic for Mini-Computer, Working Paper No. 153, M.I.T., Artificial Intelligence Laboratory, Sept. 1977.

[KOB77] N.Koblitz: P-Adic Numbers, P-Adic Analysis and Zeta Functions. Springer Verlag, New York, 1977.

[KOM81] P. Kornerup, D.W. Matula: An Integrated Rational Arithmetic Unit. Proc. 5th IEEE Symposium on Computer Arithmetic, pp. 233-240, Ann Arbor, Mich., 1981.

[KRI81] E. V. Krishnamurthy: Matrix Processor Using P-Adic Arithmetic in Theory and Practice. Academic Press, New York, 1981.

[K&M81] U.Kulish, W.L.Miranker: Computer Arithmetic in Theory and Practice. Academic Press, New York, 1981.

[MAH73] K.Mahler: Introduction to P-Adic Numbers and Their Functions. Cambridge University Press, 1973.

[MAT75] D.W.Matula: Fixed-Slash and Floating-Slash Rational Arithmetic Proc. 3rd IEEE Symposium on Computer Arithmetic, pp. 90-91, Nov. 1975.

[MIO83] A.Miola: A Unified View of Approximate Rational Arithmetics and Rational Interpolation. Proc. 6th IEEE Symposium on Comp. Arithmetic. Aarhuss, December 1983

[MIO84] A.Miola: Algebraic Approach to P-Adic Conversion of Rational Numbers. IPL 18, 1984.