Serdica J. Computing **14** (2020), No 1–2, 43–50

Serdica
Journal of Computing

Bulgarian Academy of Sciences
Institute of Mathematics and Informatics

QUADRUPLES FOR RESIDUE NUMBER SYSTEMS WITH SUM OF QUOTIENTS A POWER OF A PRIME*

Georgi Boyvalenkov, Peter Boyvalenkov

Abstract. We investigate Residue Number Systems (RNS) of special type which were recently shown to be useful for some type of computations in embedded systems. We develop an algorithm for derivation of all quadruples for RNS under investigation subject to certain restrictions.

1. Introduction. Residue number systems (RNS) are attractive for computing in digital signal processing applications because their modular design allows parallel processing in the different modular channels [1, 8, 9, 10]. In computations with RNS the implementation of non-modular operations like division, sign detection, comparison of numbers, and reverse conversion can be more effective when a diagonal function, corresponding to Sum of Quotients $SQ = 2^k$,

ACM Computing Classification System (1998): G.4, F.1.2.
Mathematics Subject Classification (2020): 68W15, 68U99.
Key words: residue number systems, sum of quotients, parallel processing.
*This work was supported in part by the Bulgarian Ministry of Education and Science by Grant No DO1-387/18.12.2020 for NCHDC, a part of the Bulgarian National Roadmap on RIs.

is used (see [11, 3] and references therein). Therefore it is important to find and classify RNS with $SQ = 2^k$.

In this note we are interested in the following more general problem—to investigate RNS with $SQ = p^k$, where p is a prime and k is a positive integer. We propose an algorithm for finding quadruples for RNS with such SQ and present and discuss some results. As a by-product, we introduce a p-ary measure for balancedness of RNS. Possible applications of such RNS will require p-ary instead of binary implementations.

We formulate the problem in Section 2 and describe our approach to it in Section 3. Section 4 is devoted to the algorithm, which follows the previous argumentation. We discuss the balancedness of the obtained quadruples and give some examples in Sections 5 and 6, respectively.

2. Residue Number Systems and their Sum of Quotients. Let $m_1, m_2, \ldots, m_n$ be mutually co-prime positive integers greater than 1, $x_i \in \{0, 1, \ldots, m_i - 1\}$, $i = 1, 2, \ldots, n$, be integers, and

$$X \equiv x_i \pmod{m_i},\ i = 1, 2, \ldots, n, \tag{1}$$

be a system of linear congruences defined by these parameters. In RNS, using the Chinese remainder theorem (see, for example, [7, Section 3.4]), a solution $X \in \{0, 1, \ldots, m_1 m_2 \ldots m_n - 1\}$ of the system (1) is associated with the n-tuple $(x_1, x_2, \ldots, x_n)$ and subsequently this n-tuple is used in operations instead of X.

Denoting

$$M = \prod_{i=1}^{n} m_i, \quad M_i = \frac{M}{m_i}, \quad i = 1, 2, \ldots, n,$$

(the number M is called *dynamical range*) one considers the sum of quotients M_i,

$$SQ := \sum_{i=1}^{n} M_i.$$

The number SQ is also called *diagonal modulus*. For example, the diagonal modulus is instrumental in the definition and the performance of the so-called *diagonal function* [5, 6]

$$D(X) := \sum_{i=1}^{n} k_i x_i \pmod{SQ},$$

where the integers $k_i \in \{1, 2, \ldots, SQ - 1\}$ are defined by the congruences

$$k_i m_i \equiv -1 \pmod{SQ}.$$

Diagonal function can be also used in RNS to binary conversion [2].

It was observed that diagonal moduli of special binary representations (very low or very high Hamming weight) are capable to provide RNS with good performance (see [11]). Classification problems for small (with $n = 3$ and 4) moduli sets resulting in diagonal modulus $SQ = 2^k - 1$ or 2^k were considered in [3]. The case $n = 6$ with $SQ = 2^k$ was considered in [4].

We are interested in the following generalization of this problem—to investigate moduli sets with Sum of quotients $SQ = p^k$, where p is a prime (fixed in advance) and k is a positive integer. Thus we search for $SQ = p^k$ with n mutually co-prime modules m_i, $i = 1, 2, \ldots, n$. Assuming that p is a fixed prime number, we need to analyze the Diophantine equation

$$p^k = \frac{M}{m_1} + \frac{M}{m_2} + \cdots + \frac{M}{m_n}, \tag{2}$$

where $M = m_1 m_2 \ldots m_n$ and $(m_i, m_j) = 1$ whenever $i \neq j$. It is easy to see that $(m_i, p) = 1$ for every $i = 1, 2, \ldots, n$.

The case $n = 2$ is trivial, since (2) is reduced then to

$$m_1 + m_2 = p^k$$

and the solutions are given by all pairs $(m_1, p^k - m_1)$ with $(m_1, p) = 1$ and $1 < m_1 < p^k$. For $n = 3$, equation (2) becomes

$$m_1 m_2 + m_2 m_3 + m_3 m_1 = p^k,$$

which has too many solutions and even a brute force approach can provide a large amount of them. Thus $n = 4$ is the first interesting case, although a complete description of the solutions seems unattainable. We will present an heuristic and computational approach with an algorithm for finding all solutions of (2) for $n = 4$ with limited modules. More precisely, our goal is to provide an algorithm for classifying all RNS quadruples (m_1, m_2, m_3, m_4) with $SQ = p^k$, where the modules m_i are bounded from above by some constant. In our implementation we set the restrictions $m_i \leq 10000$, $i = 1, 2, 3, 4$.

3. Our Approach. We shall use several times the following simple lemma. We use the notation $v_p(m)$ for the maximum power of p which divides m (i. e., the ratio $m/p^{v_p(m)}$ is an integer which is co-prime with p; $v_p(m)$ is a non-negative integer). For example, $v_2(54) = 2$, $v_3(54) = 3$, $v_5(100) = 2$, $v_5(101) = 0$.

Lemma 1. *Let u and $\ell \geq 2$ be positive integers and*

$$p^u = a_1 + a_2 + \cdots + a_\ell,$$

where $a_1, a_2, \ldots, a_\ell$ are positive integers. Then the two smallest among the non-negative integers $v_p(a_1), v_p(a_2), \ldots, v_p(a_\ell)$ are equal.

P r o o f. Assume for a contradiction (and without loss of generality) that

$$v_p(a_1) < v := \min\{v_p(a_2), \ldots, v_p(a_\ell)\}$$

(i. e., $v_p(a_1)$ is the smallest among the non-negative integers $v_p(a_1), v_p(a_2), \ldots, v_p(a_\ell)$ and it is unique with this property). Since $u > v$ (otherwise $p^u \leq p^v \leq a_2 < a_1 + a_2 + \cdots + a_\ell$), it follows that

$$0 \equiv p^u = a_1 + a_2 + \cdots + a_\ell \equiv p^{v_p(a_1)} b_1 \pmod{p^v},$$

which is impossible (here the positive integer $b_1 = a_1/p^{v_p(a_1)}$ is coprime with p). □

The first step in our investigation will be to represent the equation (2) in such a way that Lemma 1 would be effective. This can be done differently for different n and even for the same n, when n is large. In all cases we build our algorithms on the investigation of sums of two of the variables m_i.

In what follows we already focus to the case $n = 4$. The equation (2) can be written as

$$p^k = m_1 m_2 (m_3 + m_4) + m_3 m_4 (m_1 + m_2). \tag{3}$$

It follows now from Lemma 1 (with $\ell = 2$; note that m_i are coprime to p) that

$$v_p(m_1 + m_2) = v_p(m_3 + m_4) = \omega_1 \geq 0, \tag{4}$$

where ω_1 is integer. Let

$$m_1 + m_2 = p^{\omega_1} r, \quad m_3 + m_4 = p^{\omega_1} s,$$

where $(r, p) = (s, p) = 1$. Moreover, $(r, s) = 1$ because otherwise their common prime divisor will divide p^k, i. e., will be equal to p, which is impossible.

Dividing both sides of (3) by p^{ω_1}, we obtain the equation

$$p^{k-\omega_1} = r m_3 m_4 + s m_1 m_2,$$

whence we express

$$m_3 m_4 = \frac{p^{k-\omega_1} - s m_1 m_2}{r}.$$

Hence, for fixed $m_1 + m_2 = A$ and $m_3 + m_4 = B$ with $v_p(A) = v_p(B) = \omega_1$ the numbers m_3 and m_4 are the roots of the quadratic equation

$$X^2 - BX + C = 0,$$

where

$$C = \frac{p^{k-\omega_1} - s m_1 m_2}{r} = m_3 m_4.$$

This representation is instrumental in our algorithm below since it allows tangibly faster determination of the feasible pairs (m_3, m_4) (compared to brute force by embedded cycles of the sums $m_1 + m_2 = A$ and $m_3 + m_4 = B$). We run all possible powers $p^{\omega_2} = p^{k-\omega_1}$ in order to find these which are divisible by r.

In the above scheme, the following restrictions can be applied:

$$A \le B \le 2c_4,\ C \le B^2/4,$$

r divides $p^{k-\omega_1} - s m_1 m_2$, $B^2 - 4C$ is a perfect square less than B^2, and

$$m_{3,4} = \frac{-B \pm \sqrt{B^2 - 4C}}{2}$$

are integers in the interval $[2, c_4]$, coprime with p. Our implementation uses $c_4 = 10000$.

4. An Algorithm for Finding Quadruples. The above analysis can be turned into classification as follows.

Step 1. For fixed $A = m_1 + m_2 \in [5, 2c_4]$ consider all sums $B = m_3 + m_4 \in [A, 2c_4]$ such that the condition (4) is satisfied. For each pair (A, B) compute $r = A/p^{\omega_1}$ and $s = B/p^{\omega_1}$.

Step 2. For every pair (A, B) from Step 1, consider all pairs (m_1, m_2) with sum A, $(m_1, m_2) = (m_1, p) = (m_2, p) = 1$, and for every such pair search for feasible powers of p, say p^{ω_2}, such that r divides $p^{\omega_2} - s m_1 m_2$. Compute $C = (p^{\omega_2} - s m_1 m_2)/r$.

Step 3. For any quadruple (ω_2, m_1, m_2, B) from Step 2, check whether the roots (they will be m_3 and m_4) of the equation $X^2 - BX + C = 0$ are integers in the interval $[2, c_4]$, which are mutually coprime and coprime to p.

Output: p, (m_1, m_2, m_3, m_4), $k = \omega_1 + \omega_2$, $SQ = p^k$.

Theorem 1. *The so chosen m_1, m_2, m_3, m_4 form an RNS quadruple with $SQ = p^k$.*

Proof. It is clear from the above that every quadruple (m_1, m_2, m_3, m_4) obtained this way has $SQ = p^k$. Moreover, since the search is exhaustive with respect to the pairs (m_1, m_2) and the consequent derivation of m_3 and m_4 is uniquely determined in the chosen limits, all such quadruples are found. □

5. Balancedness of Our RNS Quadruples. A measure for balancedness of RNS was proposed in [4]. Here we consider its p-ary version as follows.

Let (m_1, m_2, m_3, m_4) be an RNS quadruple (with or without $SQ = p^k$). Denote $b_i := \lceil \log_p m_i \rceil$, $i = 1, 2, 3, 4$ (of course, this can be generalized for any number of modules) and

$$\bar{b} = \frac{b_1 + b_2 + b_3 + b_4}{4}.$$

Then

$$\beta := \sum_{i=1}^{4} (\bar{b} - b_i)^2$$

shows the deviation of the quadruple (b_1, b_2, b_3, b_4) from the "ideal" quadruple of equal "widths" b_i. It is natural to expect that quadruples with smaller β will perform better in most applications. Of course, the most interesting[1] case comes with $b_1 = b_2 = b_3 = b_4 = \bar{b}$, i. e., $\beta = 0$.

It is clear that it is easy to construct unrestricted RNS with $\beta = 0$. However, the addition of the condition $SQ = p^k$ makes the problem for finding well balanced RNS quite difficult. As we will see in the next section, there are only two quadruples with $\beta = 0$ for $p = 2$ and none for $p = 3$ (with moduli less than 10000).

6. Some Results. We implemented the above algorithm by C++ for each prime less than 100 with restrictions $m_i \leq 10000$, $i = 1, 2, 3, 4$. In the section we describe some results for $p = 2$ and $p = 3$.

6.1. $p = 2$ There are 165 quadruples with Sum of Quotients $SQ = 2^k$ and $m_i \leq 10000$, $i = 1, 2, 3, 4$. The smallest dynamical range M is 4220805 (by the quadruple $(5, 29, 93, 313)$) and the largest one is $\approx 1,768.10^{15}$ (see below). The parameter k is always even, between 20 and 40.

[1] If $\beta = 0$, all parallel computational channels will have the same width. Of course, this can be done for $\beta > 0$, but there will be less efficient.

Only two of these quadruples,

$$(5377, 6253, 7209, 7297), \quad M \approx 1,768.10^{15}, \quad k = 40,$$
$$(5129, 6041, 6921, 8173), \quad M \approx 1,752.10^{15}, \quad k = 40,$$

have balancedness parameter $\beta = 0$. These two quadruples have the largest dynamical range M among all obtained.

Then come four quadruples with the next possible $\beta = 9/16 = 0.75$, namely

$$(641, 1025, 1249, 1269), \quad M \approx 1,041.10^{12}, \quad k = 32,$$
$$(1559, 2667, 2987, 3355), \quad M \approx 4,166.10^{13}, \quad k = 36,$$
$$(2167, 3943, 4039, 7235), \quad M \approx 2,496.10^{14}, \quad k = 38,$$
$$(2533, 4353, 4421, 5453), \quad M \approx 2,658.10^{14}, \quad k = 38.$$

The whole list of all 165 quadruples obtained is available from the second author upon request.

6.2. $p = 3$ There are 229 quadruples with $SQ = 3^k$ and $m_i \leq 10000$, $i = 1, 2, 3, 4$, with dynamical range M between 31450 and $\approx 1,192.10^{15}$. The parameter k takes all values between 9 and 25 except 10.

The smallest β is $9/16 = 0,75$, attained by 9 quadruples. One of them,

$$(3457, 6985, 6986, 6989), \ M \approx 1,178.10^{15}, \ k = 25,$$

is remarkable with its three very close entries.

The whole list of all 229 quadruples with $SQ = 3^k$ is available from the second author upon request.

REFERENCES

[1] Ananda Mohan P. V. Residue Number Systems: Algorithms and Architectures. Kluwer, Dordrecht, 2002.

[2] Ananda Mohan P. V. RNS to Binary Conversion Using Diagonal Function and Pirlo and Impedovo Monotonic Function. *Circuits Syst. Signal Process*, **35** (2015), 8–13.

[3] Boyvalenkov P., N. Chervyakov, P. Lyakhov, N. Semyonova, A. Nazarov, M. Valueva, G. Boyvalenkov, D. Bogaevskiy, D. Kaplun. Classification of Moduli Sets for RNS with Special Diagonal Functions. *IEEE Access*, **8** (2020), 156104–156116.

[4] Boyvalenkov P., P. Lyakhov, N. Semyonova, M. Valueva, G. Boyvalenkov, A. Voznesensky, D. Kaplun. Residue Number Systems with Six Modules and Efficient Circuits Based on Power-of-two Diagonal Modulus, submitted 2021.

[5] Dimauro G., S. Impedevo, G. Pirlo. A New Technique for Fast Number Comparison in the Residue Number System. *IEEE Trans. Comput.*, **42** (1993), 608–612.

[6] Dimauro G., S. Impedevo, G. Pirlo, A. Salzo. RNS Architectures for the Implementation of the Diagonal Function. *Inf. Process. Lett.*, **73** (2000), 189–198.

[7] Ireland K., M. Rosen. A Classical Introduction to Modern Number Theory. 2nd ed., New York, Springer-Verlag, 1990.

[8] Molahosseini A. S., L. S. de Sousa, C.-H. Chang (eds). Embedded Systems Design with Special Arithmetic and Number Systems. Springer, 2017.

[9] Omondi A. R., B. Premkumar. Residue Number Systems: Theory and Implementation. Imperial College Press, London, 2007.

[10] Soderstrand M. A., G. A. Jullien, W. K. Jenkins, F. Taylor (eds). Residue Number System Arithmetic: Modern Applications in Digital signal Processing. IEEE Press, London, 1986.

[11] Valueva M., G. Valuev, N. Semyonova, P. Lyakhov, N. Chervyakov, D. Kaplun, D. Bogaevskiy. Construction of Residue Number System using Hardware Efficient Diagonal Function. *Electronics*, **8** (2019), No 6, Article 694.

Georgi Boyvalenkov
Bosch.IO LTD
47B Tsarigradsko shosse
1124 Sofia, Bulgaria
e-mail:
`georgi.boyvalenkov@bosch.io`

Peter Boyvalenkov
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str., Bl. 8
1113 Sofia, Bulgaria
e-mail: `peter@math.bas.bg`

Received July 22, 2021
Final Accepted October 5, 2021