Neurophysiological Discoveries of the 2014 Nobel Prize Winners in Medicine from a Computer Arithmetic Perspective

Behrooz Parhami

Dept. Electrical & Computer Engr.
Univ. of California, Santa Barbara, USA

parhami@ece.ucsb.edu

Presented at the 20th Int’l Symp. Computer Architecture and Digital Systems
Guilan University, Iran, August 2020

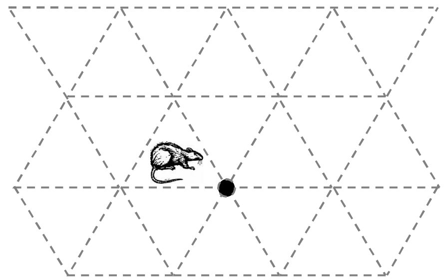
About This Presentation

A preliminary version of this slide show was developed for a talk at Asilomar Conference on Signals, Systems, and Computers, held November 1-4, 2009. Subsequently, the slide show was updated and expanded to incorporate results of new research.

Edition	Released	Revised	Revised	Revised
First	Nov. 2009	June 2019	Aug. 2020	

Outline

- Introduction / Background
  - What were the discoveries?
  - Mixed digital/analog arithmetic
  - Residue number system (RNS)
- RNS with Continuous Digits (CD-RNS)
  - Distinct from conventional RNS
  - Motivations for this study
- Dynamic Range and Precision
- Choosing the CD-RNS Moduli
- Conclusions / Future Work



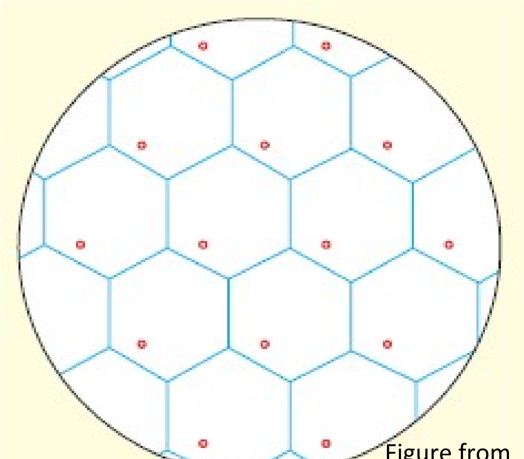
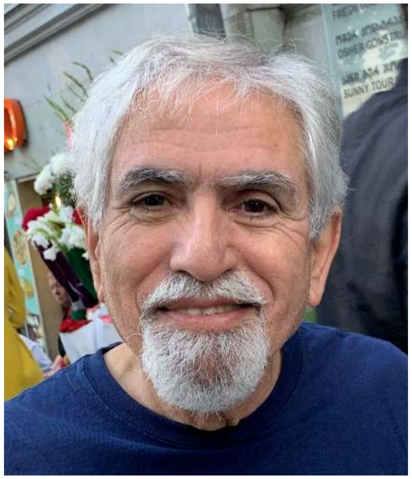


Figure from Current Biology

Abstract

The discovery that mammals use a multi-modular method akin to residue number system (RNS), but with continuous residues or digits, to encode position information led to the award of the 2014 Nobel Prize in Medicine. After a brief review of the evidence in support of this hypothesis, and how it relates to RNS, I discuss the properties of continuous-digit RNS, and present results on the dynamic range, representational accuracy, and factors affecting the choice of the moduli, which are themselves real numbers. I conclude with suggestions for further research on important open problems concerning the process of selection, or evolutionary refinement, of the set of moduli in such a representation.

Speaker's Brief Technical Bio



Behrooz Parhami (PhD, UCLA 1973) is Professor of Electrical and Computer Engineering, and former Associate Dean for Academic Personnel, College of Engineering, at University of California, Santa Barbara, where he teaches and does research in the field of computer architecture: more specifically, in computer arithmetic, parallel processing, and dependable computing.

A Life Fellow of IEEE, a Fellow of IET and British Computer Society, and recipient of several other awards (including a most-cited paper award from *J. Parallel & Distributed Computing*), he has written six textbooks and more than 300 peer-reviewed technical papers. Professionally, he serves on journal editorial boards (including for 3 different *IEEE Transactions*) and conference program committees, and he is also active in technical consulting.

How Looking at Nature Helps my Research

Parallel processing

Parallelism used extensively in human brain and other natural systems

Dependable (fault-tolerant) computing

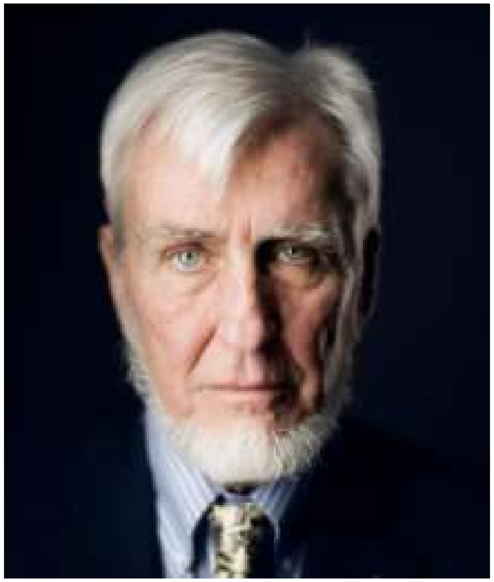
The self-healing amphibian axolotl can regenerate a near-perfect replica of almost any body part it loses

Computer arithmetic

My subject area today: Use of residue representation in rat's navigational system

Nobel Prize in Physiology or Medicine: 2014

One half went to John O'Keefe (University College, London), the other half to May-Britt Moser (Center for Neural Computation, Norway) and Edvard I. Moser (Kavli Institute for Systems Neuroscience, Norway) "for their discoveries of cells that constitute a positioning system in the brain."

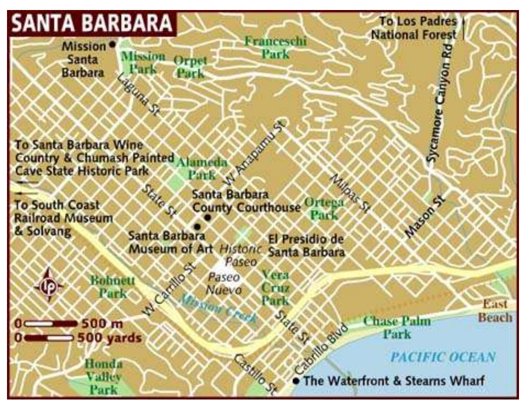


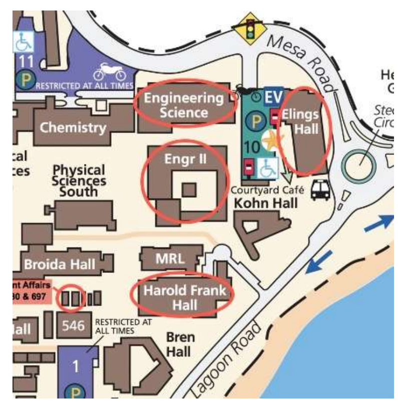
Sense of Place in Humans and Animals

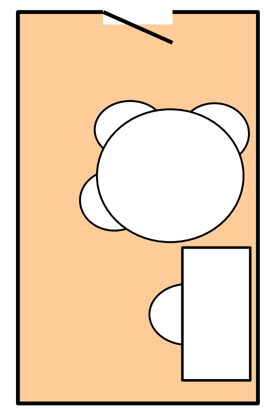
The sense of place and the ability to navigate are some of the most fundamental brain functions.

German philosopher Immanuel Kant (1724-1804) argued that some mental abilities exist independent of experience.

He considered perception of place as one of these innate abilities through which the external world had to be organized/perceived.





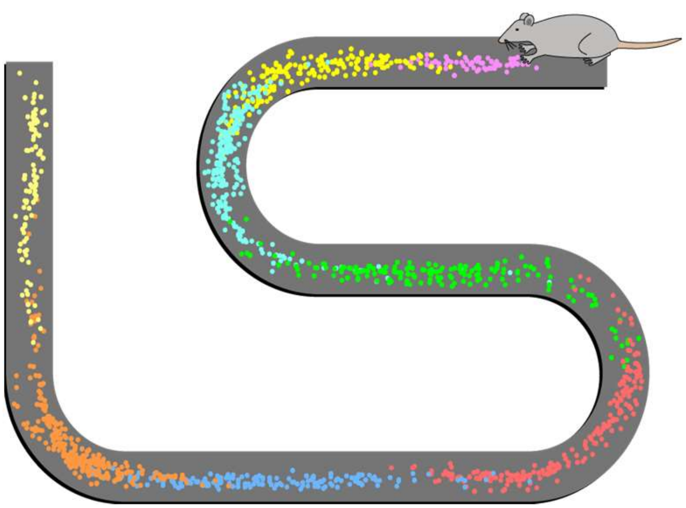


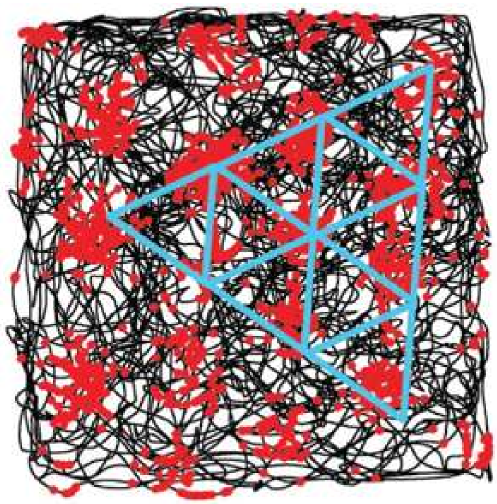
The Nobel Laureates’ Contributions

John O’Keefe discovered place cells in the hippocampus that signal position and provide the brain with spatial memory capacity.

May-Britt Moser and Edvard I. Moser discovered in the medial entorhinal cortex, a region of the brain next to hippocampus, grid cells that provide the brain with a coordinate system for navigation.

Place cells firings (image from Wikipedia)





Grid cells firings (image from Moser/Rowland/Moser, 2015)

First Attempt at Understanding

6858 • The Journal of Neuroscience, July 2, 2008 • 28(27):6858–6871

Behavioral/Systems/Cognitive

What Grid Cells Convey about Rat Location

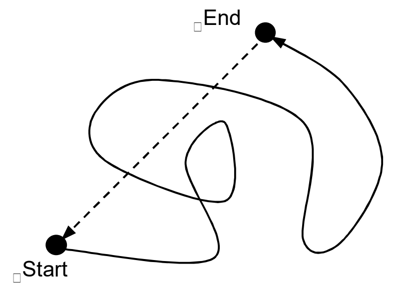
Ila R. Fiete,[1,3] Yoram Burak,[1,4] and Ted Brookings[2,5]

[1]Kavli Institute for Theoretical Physics and [2]Department of Physics, University of California, Santa Barbara, Santa Barbara, California 93106, [3]Broad Fellows Program, Computation and Neural Systems, California Institute of Technology, Pasadena, California 91125, [4]Swartz Fellows Program, Center for Brain Science, Harvard University, Cambridge, Massachusetts 02138, and [5]Volen Center, Department of Biology, Brandeis University, Waltham, Massachusetts 02454

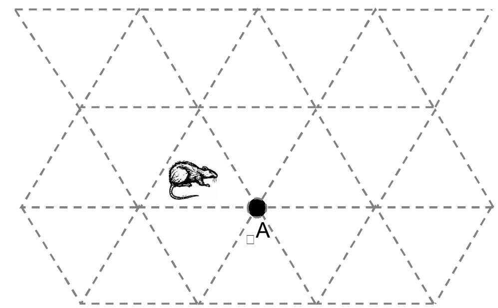
We characterize the relationship between the simultaneously recorded quantities of rodent grid cell firing and the position of the rat. The formalization reveals various properties of grid cell activity when considered as a neural code for representing and updating estimates of the rat's location. We show that, although the spatially periodic response of grid cells appears wasteful, the code is fully combinatorial in capacity. The resulting range for unambiguous position representation is vastly greater than the ≈1–10 m periods of individual lattices, allowing for unique high-resolution position specification over the behavioral foraging ranges of rats, with excess capacity that could be used for error correction. Next, we show that the merits of the grid cell code for position representation extend well beyond capacity and include arithmetic properties that facilitate position updating. We conclude by considering the numerous implications, for downstream readouts and experimental tests, of the properties of the grid cell code.

Localization with Grid Cells

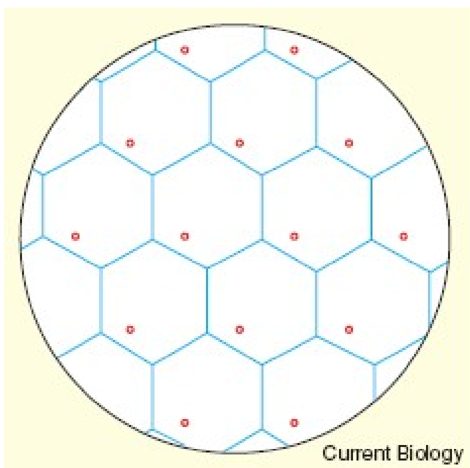
- Rat's navigation system
  - Wavy travel path
  - Straight return path
  - Even in the dark
- Nervous system has place cells & grid cells
  - Grid cell firings
  - Relative in-cell position
- In-cell positions within several grids pinpoints rat's absolute location



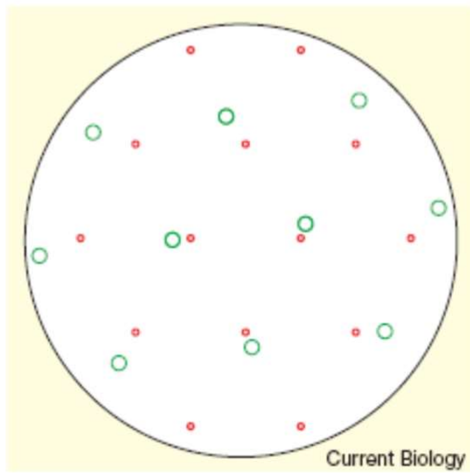
(a) Travel and return paths



(b) Rat's hexagonal grid



(c) Firings and locations [3]



(d) Two hexagonal grids [3]

The Questions to Be Addressed

- A rat can go up to a certain distance and still be able to find its way back (range)
  - Translating grid-cell firings to spatial information
  - How the range is related to grid-cell parameters
  - Representation range vs. the observed distance
- Fiete, Burak, and Brookings had connected the grid cells to residue representation
  - Couldn't confirm the hypothesis theoretically
  - Relied on extensive simulation for confirmation

My First Contribution to the Problem

Advance Access publication on 17 July 2014
doi:10.1093/comjnl/bxu060

Digital Arithmetic in Nature: Continuous-Digit RNS

BEHROOZ PARHAMI*

University of California, Santa Barbara, CA 93106-9560, USA
**Corresponding author: parhami@ece.ucsb.edu*

It has been reported in the literature on computational neuroscience that a rat's uncanny ability to dash back to a home position in the absence of any visual clues (or in total darkness, for that matter) may stem from its distinctive method of position representation. More specifically, it is hypothesized that the rat uses a multimodular method akin to residue number system (RNS), but with continuous residues or digits, to encode position information. After a brief review of the evidence in support of this hypothesis, and how it relates to RNS, we discuss the properties of continuous-digit RNS, and derive results on the dynamic range, representational accuracy and factors affecting the choice of the moduli, which are themselves real numbers. We conclude with suggestions for further research on important open problems concerning the process of selection, or evolutionary refinement, of the set of moduli in such a representation.

RNS with Analog Digits (Remainders)

- I formulated the spatial representation problem with the grid cells to CD-RNS
  - First time RNS is used with analog remainders
  - Conventional RNS theory is inapplicable
  - I developed a theory for CD-RNS and its range
- Analog and mixed digital-analog technology has a long history in computer arithmetic
  - Brief review presented in the next few slides
  - More use of analog features expected to come

Quasi-Digital Parallel Counter

- Analog current summing
  - 7 inputs, 3-bit output
  - (*): Number of 1 inputs required to produce a 1
- The scheme is even older
  - Riordan and Morton, Use of Analog Techniques in Binary Arithmetic Units, *IEEE TC*, Feb. 1965

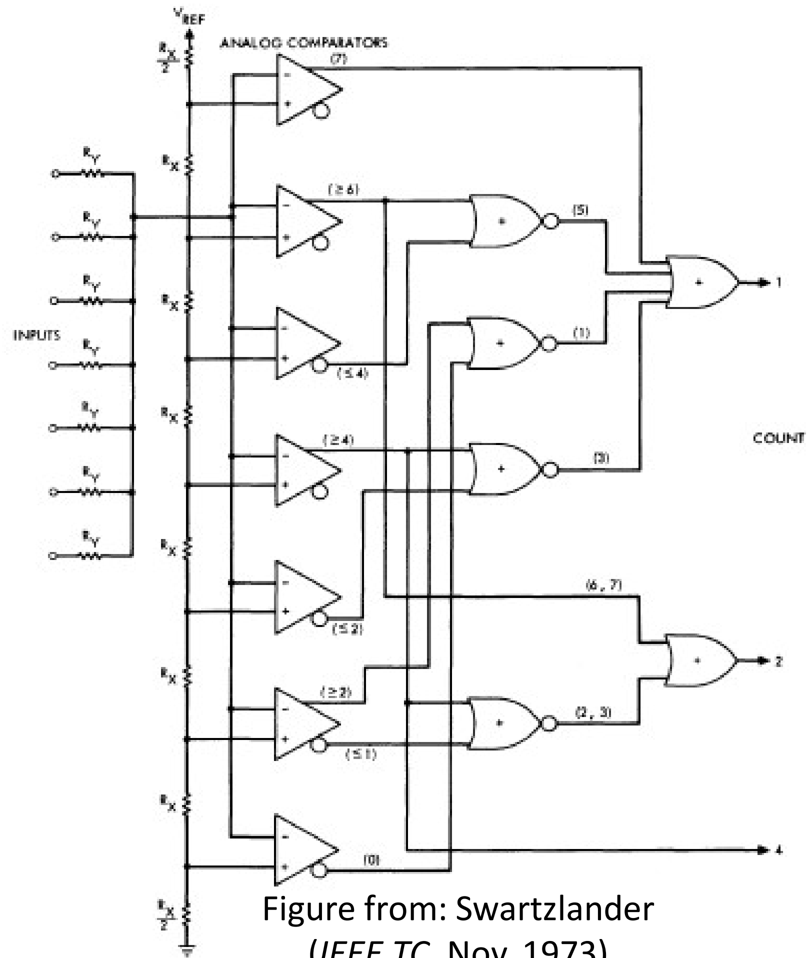
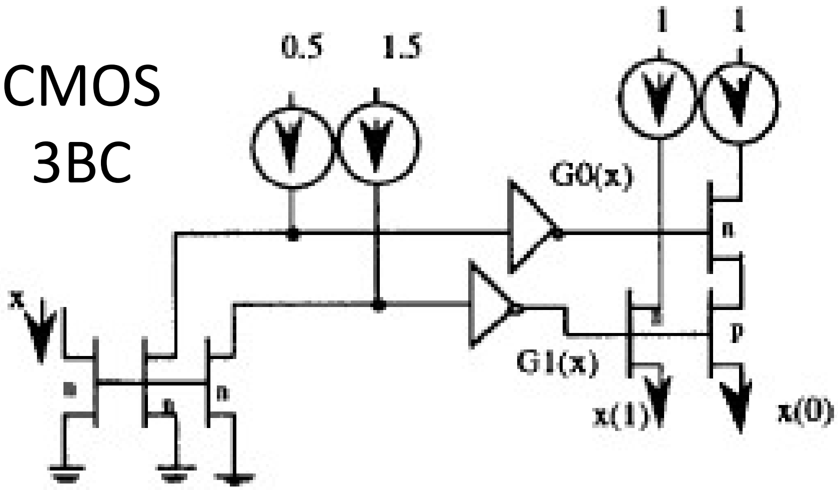


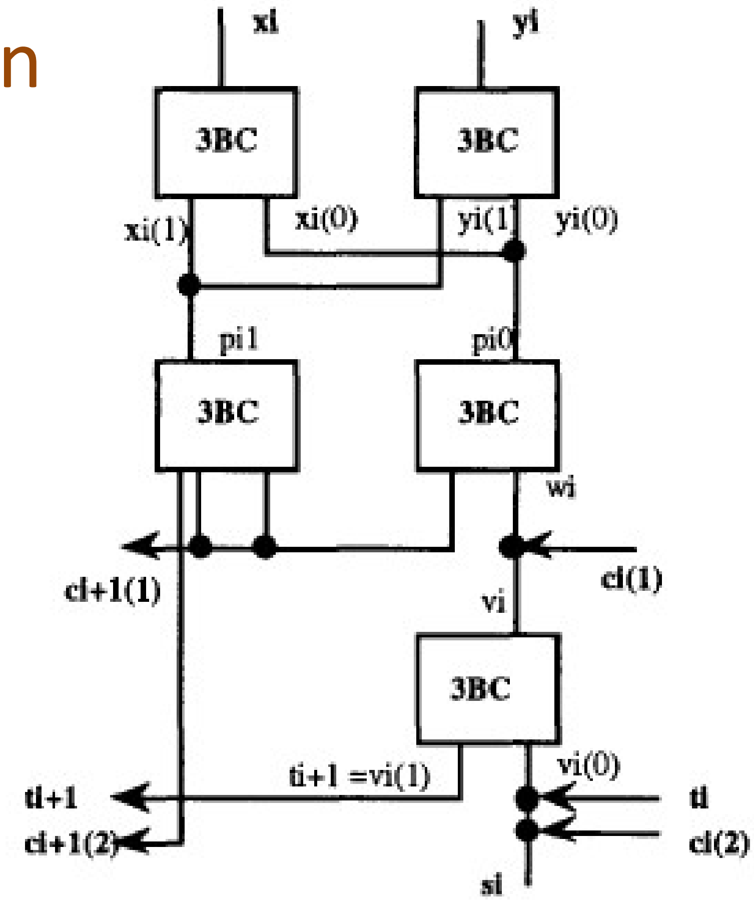
Figure from: Swartzlander (*IEEE TC*, Nov. 1973)

Current-Summing Multivalued Logic

- Binary stored-carry addition
  - Limited-carry algorithm
- 3-valued to binary conv.: 3BC

CMOS 3BC





Figures from: Etiemble & Navi (*SMVP*, May 1993)

Mixed D/A Positional Representation

- Continuous-valued number system (CVNS)
  - The MSD has all the magnitude info
  - Other digits provide successive refinements
- Familiar example: utility meter

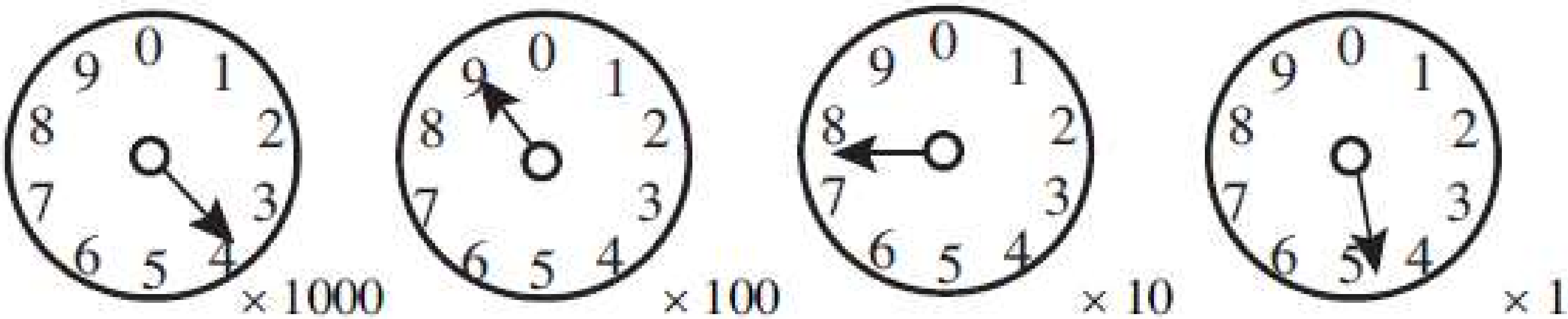


Figure from: Saed, Ahmadi, Jullien (*IEEE TC*, 2002)

An Ancient Chinese Puzzle

Puzzle, due to the Chinese scholar Sun Tzu, 1500^+ years ago:

What number has the remainders of 2, 3, and 2 when divided by 7, 5, and 3, respectively?

Residues (akin to digits in positional systems) uniquely identify the number, hence they constitute a representation: $(2 \mid 3 \mid 2)_{RNS(7|5|3)}$

In a weird way, RNS is a weighted representation

For RNS(7 | 5 | 3), the weights of the 3 positions are:

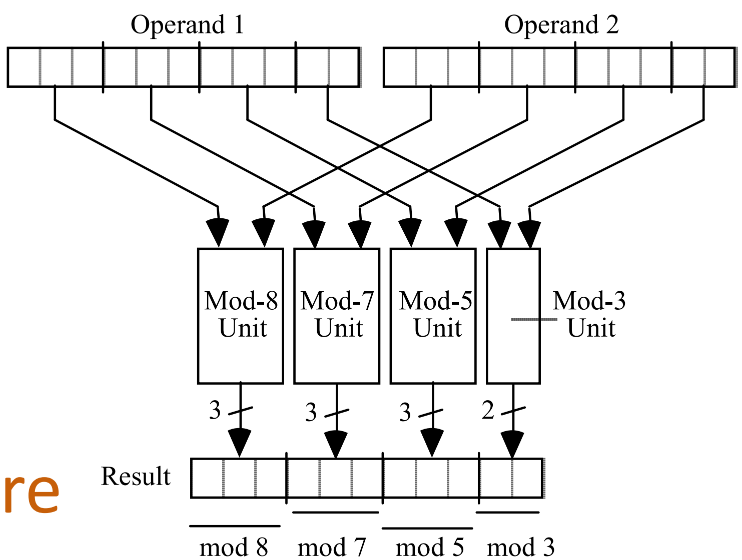
15 **21** **70**

Example -- Chinese puzzle: $(2 \mid 3 \mid 2)_{RNS(7|5|3)}$ represents the number

$$\langle \mathbf{15} \times 2 + \mathbf{21} \times 3 + \mathbf{70} \times 2 \rangle_{105} = \langle 233 \rangle_{105} = 23$$

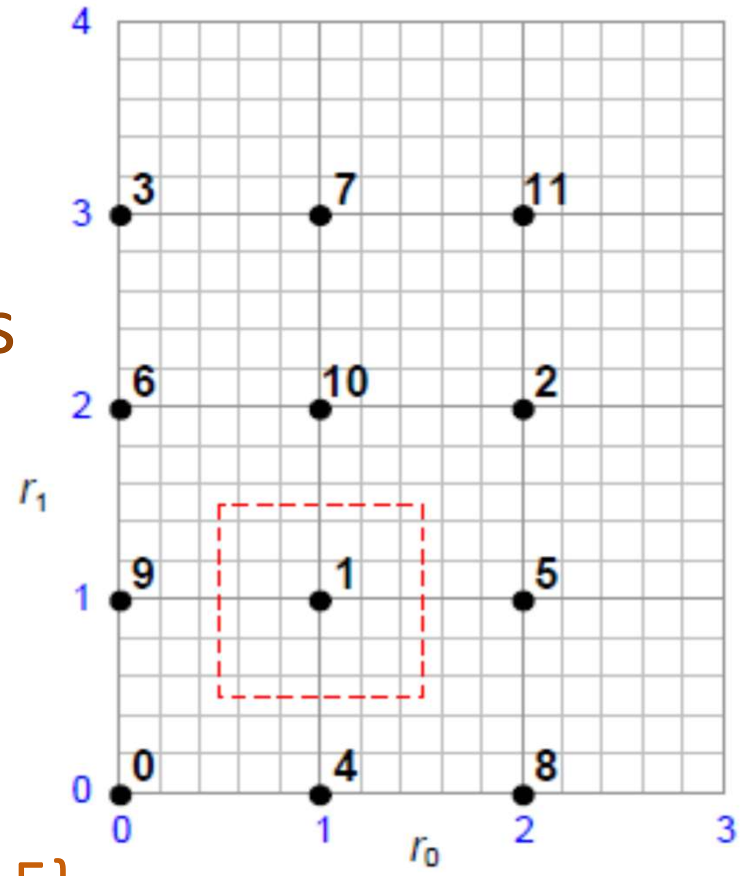
Residue Number System (RNS)

- Pairwise prime moduli: $m_{k-1} > \ldots > m_1 > m_0$
- Representation of x: $\{r_i = x \bmod m_i \mid 0 \le i \le k-1\}$
- RNS dynamic range: $M = \prod_{0 \le i \le k-1} m_i$
  - Unsigned in $[0, M-1]$
  - Signed in $[-M/2, M/2-1]$
- RNS arithmetic algorithms
  - Digitwise add, sub, mult
  - Difficult div, sign test, compare



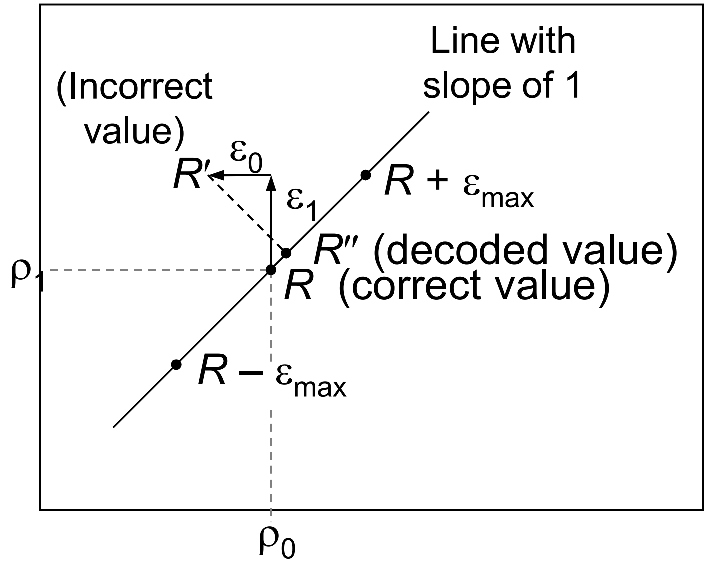
Integer Moduli and Residues

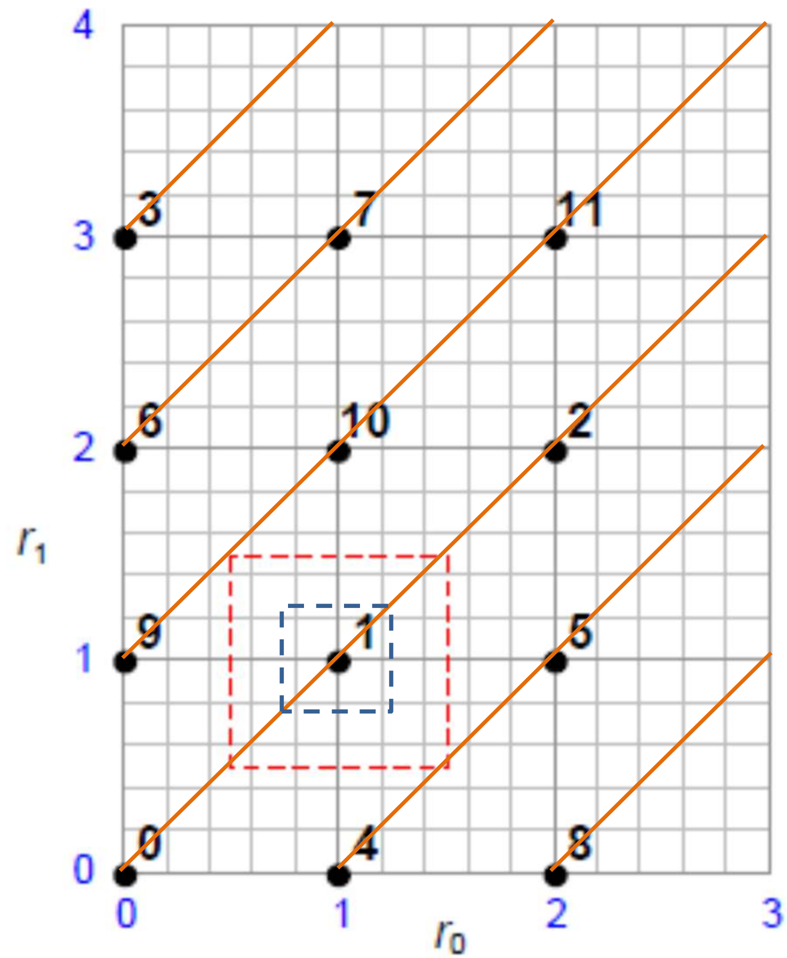
- Two-modulus RNS {4, 3}
- Dynamic range [0, 11]
- Imagine residues with errors
  - Errors < 0.5 correctable
  - Errors < 1.0 detectable
- Multiresidue systems
  - 3-modulus RNS {5, 4, 3}
  - {5,4,3} ≡ {20,3} ≡ {15,4} ≡ {12,5}



Integer Moduli, Continuous Residues

- Residue errors ε_1 and ε_0
- Decoding error $\leq \max(\varepsilon_1, \varepsilon_0)$
- Dynamic range? $[0, 12 - \varepsilon_{max}]$
- Max allowable error < 0.25





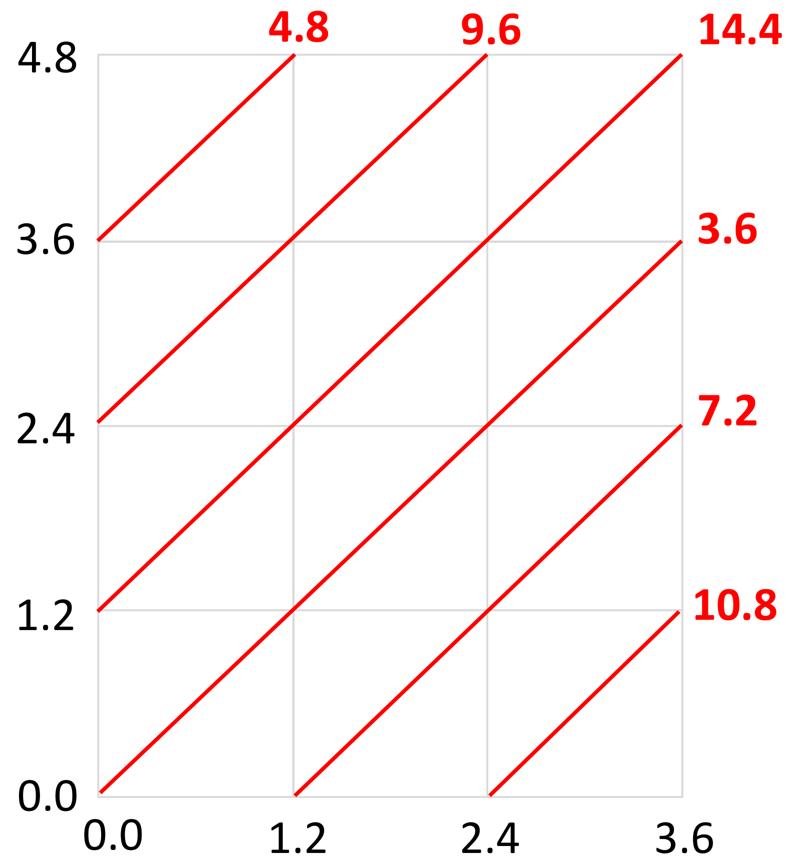
Continuous Moduli and Residues

Case 1: The moduli are integer multiples of their difference

With proper scaling, the CD-RNS can be converted to an RNS

This example is equivalent to RNS {4, 3} with scale factor 1.2

Question:
Are there CD-RNSs that cannot be replaced with ordinary RNSs?

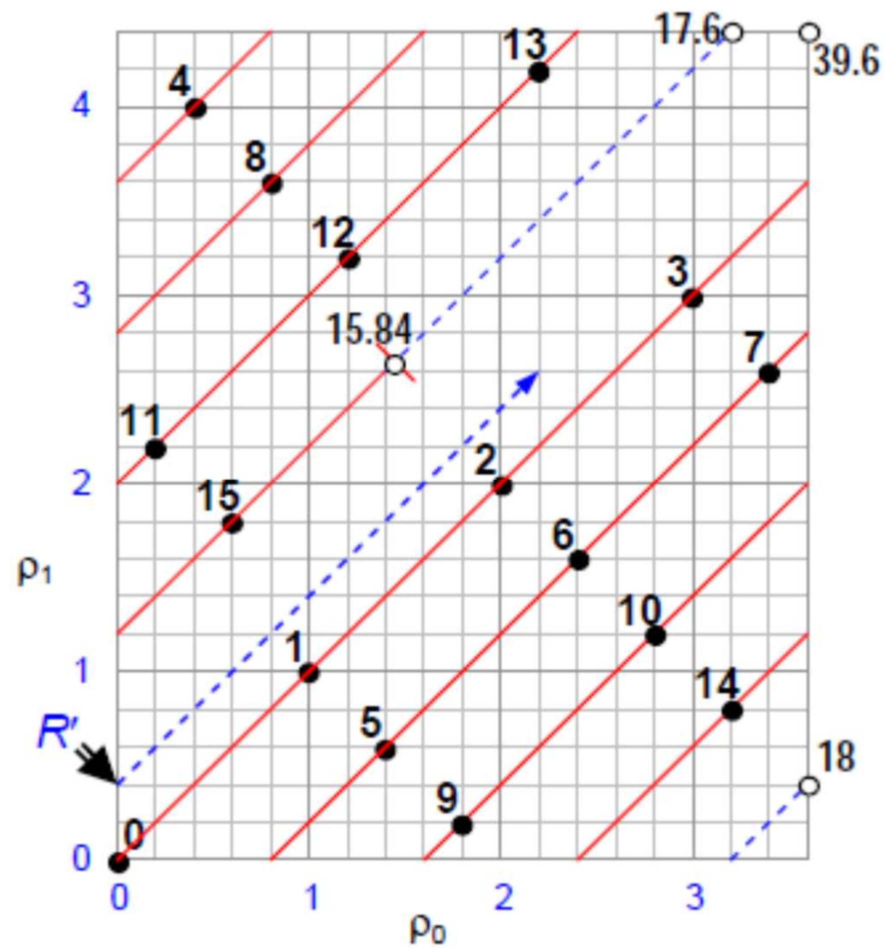


Equivalence of CD-RNS and RNS

Case 2a: The moduli are integer multiples of some number *s* (that divides their difference)

With proper scaling, the CD-RNS can be converted to an RNS, provided max error target is ≤ *s*/4

For this example, *s* = 0.4 and the system is equivalent to RNS {11, 9} with scale factor 0.4

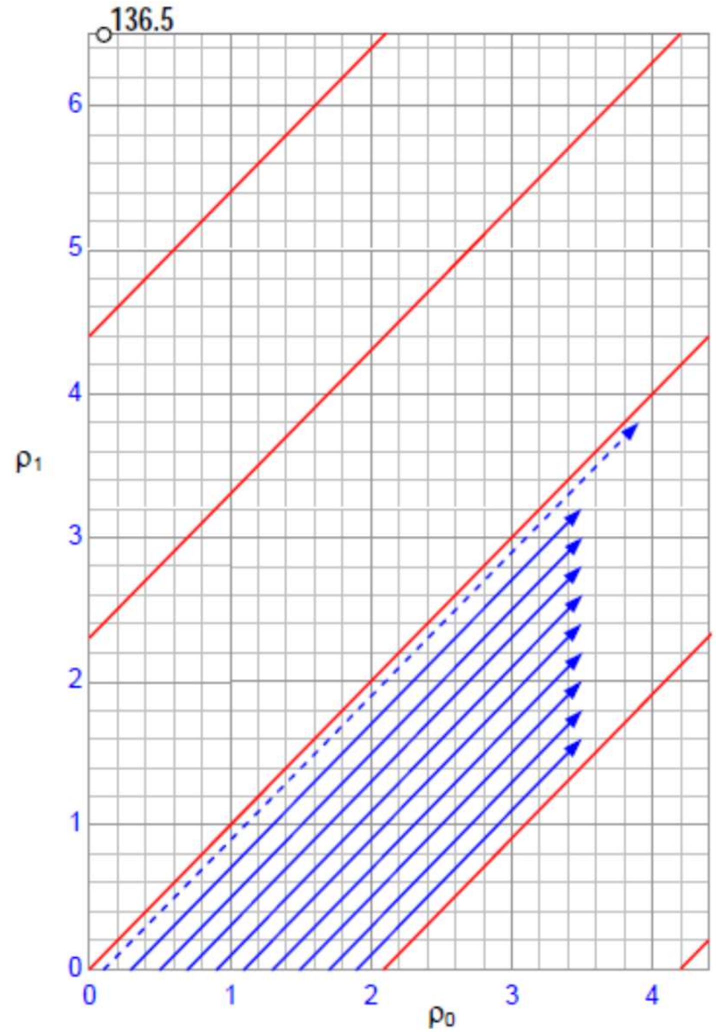


Representational Power of CD-RNS

Case 2b: The moduli are integer multiples of some number *s* (that divides their difference), but max error target > *s*/4

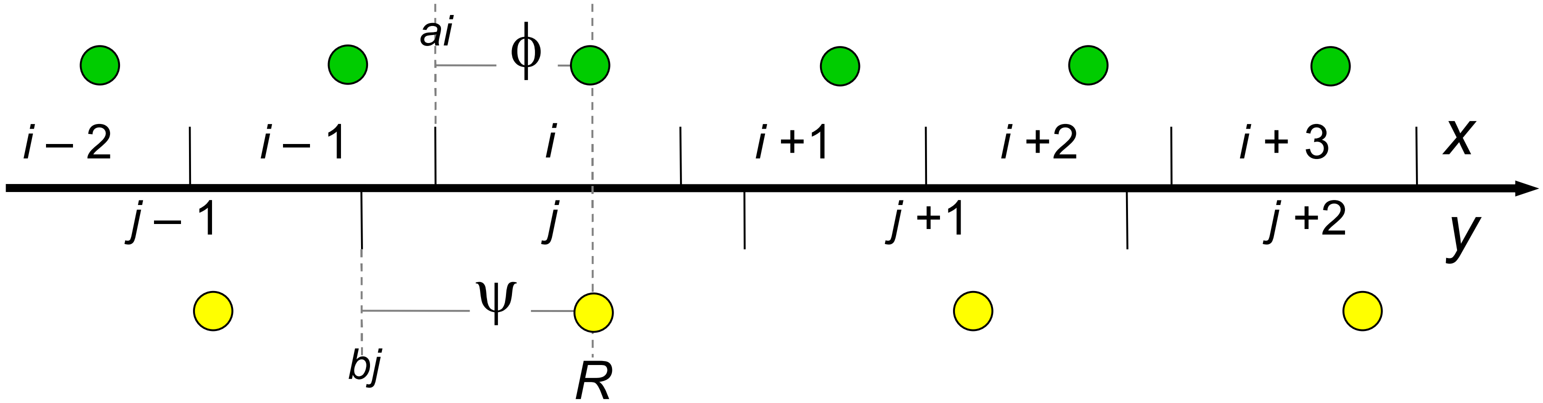
The CD-RNS is not equivalent to an RNS in terms of representational capability and dynamic range

For this example, *s* = 0.1 but the system is different from RNS {65, 44} with scale factor 0.1



Conceptually Simpler 1D Example

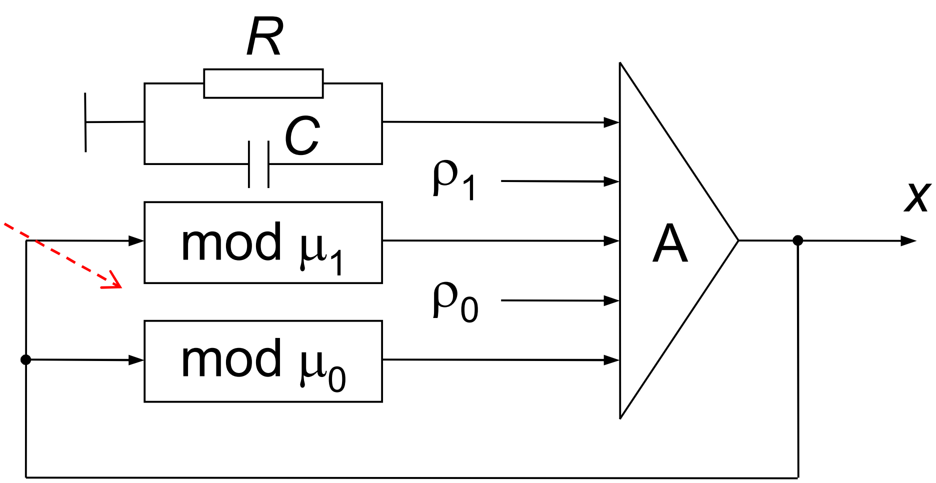
- Distance encoded by mod-*a* and mod-*b* residues
  - Phases ϕ and ψ given
  - Reverse conversion provides *R*
- *R* is a point whose mod-*a* and mod-*b* residues match ϕ and ψ to within the error bound



Backward Conversion to Binary

- CRT and its derivatives are inapplicable
  - Conversion amplifies the errors
  - Example 15 in my 2015 *Computer Journal* paper
- View the conversion as nonlinear optimization
  - Convergence occurs with circuit's RC time constant

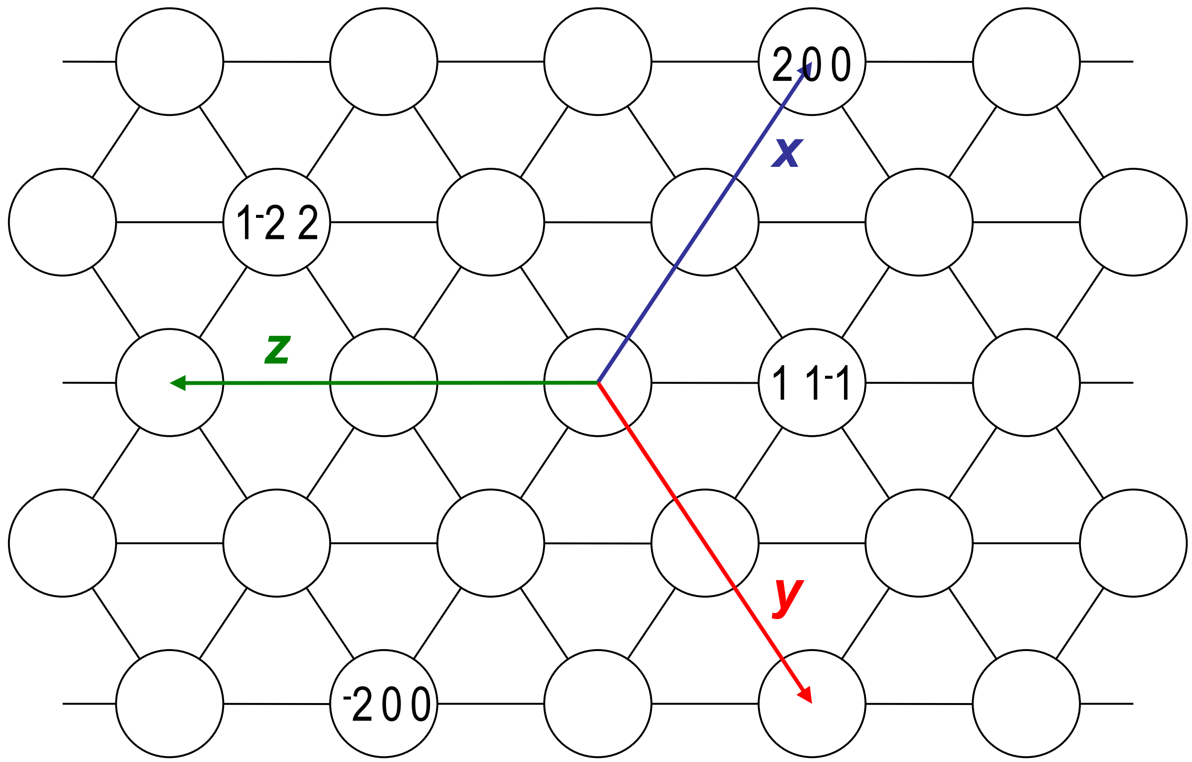
Forward conversion



Sun & Yao, *IEEE Int'l Conf. Neural Networks*, 1994

Hex Grid Coordinate System

- Point identified by 3 coordinates, one of which is redundant
- Redundancy allows error correction beyond the system's accuracy range



Open Problems in Neurobiology

- Dynamic range of rat's navigation system
- Numerical simulation: Range $\sim (1/\varepsilon_{max})^{Exponent}$

  Exponent $\approx$ Number of moduli $- \theta$
- **Example:** 12 moduli $\Rightarrow$ Exponent = 10.7

  Our results yield an exponent of 11.0
- How did the rat's navigational grids evolve?
  (Evolutionary basis for moduli optimization)

Dynamic Range Lower Bound

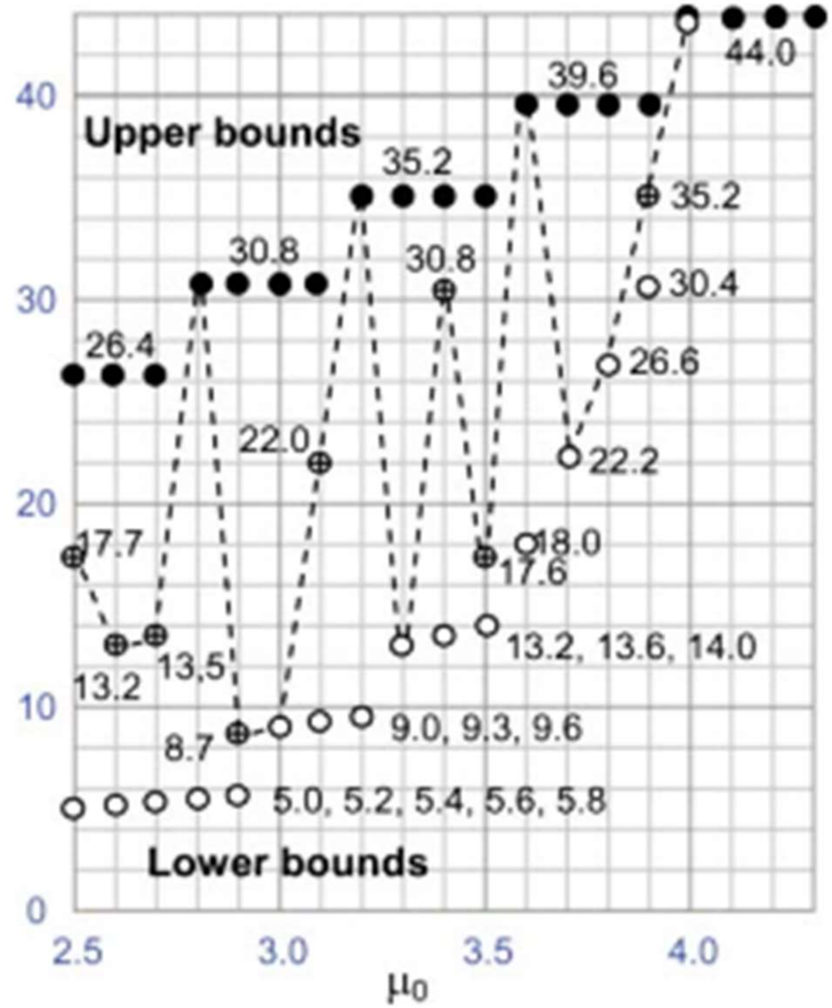
- CD-RNS with the moduli μ_1 and μ_0
- $\sigma_{-1} = \mu_1$; $\sigma_0 = \mu_0$; $\sigma_{i+1} = \min(|\sigma_{i-1}|_{\sigma i}, \sigma_i - |\sigma_{i-1}|_{\sigma i})$
- **Theorem 2:** Dynamic range is at least $\mu_0(1 + \lfloor\mu_1/\mu_0\rfloor\lfloor\mu_0/\sigma_1\rfloor)\lfloor\sigma_1/\sigma_2\rfloor\lfloor\sigma_2/\sigma_3\rfloor \ldots \lfloor\sigma_{j-1}/\sigma_j\rfloor$ where j is the largest index for which $\sigma_j \geq 2\varepsilon_{max}$
- **Intuition:** Remove floors to get $\mu_0\mu_1/(2\varepsilon_{max})$
- **Example 6:** CD-RNS with $\mu_1 = 4.4$, $\mu_0 = 3.6$, $\varepsilon_{max} = 0.2$ $\Rightarrow$ $\sigma_1 = 0.8$, $\sigma_2 = 0.4$ $\Rightarrow$ Dynamic range ≥ 36.0

Dynamic Range Upper Bound

- CD-RNS with the moduli μ_1 and μ_0
- δ = Largest number that divides μ_1 and μ_0 if it exists, 0 otherwise
- **Theorem 3:** Dynamic range is at most
  $\max(\mu_0 \lfloor \mu_1/\gamma \rfloor, \mu_1 \lfloor \mu_0/\gamma \rfloor)$
  where $\gamma = \max(2\varepsilon_{max}, \delta)$
- **Intuition:** Remove floors to get $\mu_0\mu_1/\gamma$
- **Example 6:** CD-RNS with $\mu_1 = 4.4$, $\mu_0 = 3.6$, $\varepsilon_{max} = 0.2$
  $\Rightarrow \delta = 0.4, \gamma = 0.4 \Rightarrow$ Dynamic range ≤ 39.6

Lower and Upper Bounds Example

- Example 10 in paper
- Fix μ_1 at 4.4
- Vary μ_0 in steps of 0.1
- Range varies (dashed)
- Tightness varies
- Matching of upper bound = Optimality?
- Achieving wider range

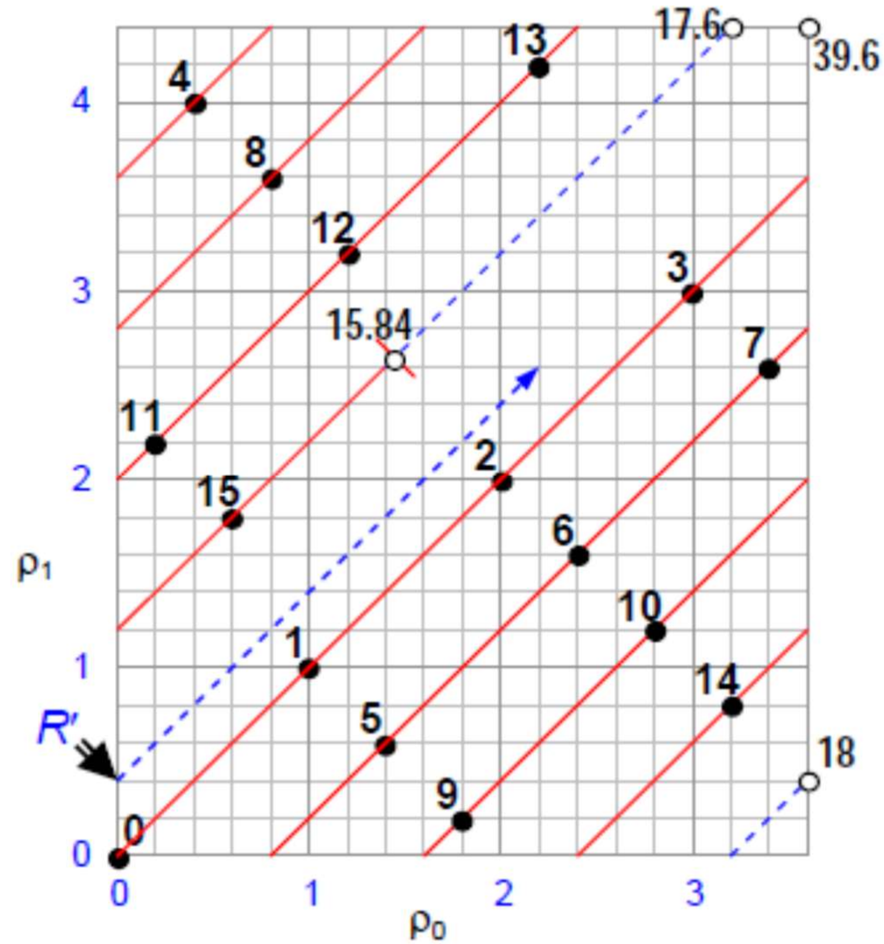


Choosing the CD-RNS Moduli

Theorem 2: $\mu \geq 36.0$
Theorem 3: $\mu \leq 39.6$

Intuitively, the moduli are optimal when the two bounds coincide

To cover the dynamic range μ, choose the moduli that are on the order of $(2\mu\varepsilon_{max})^{1/2}$ and differ by $2\varepsilon_{max}$



Conclusions

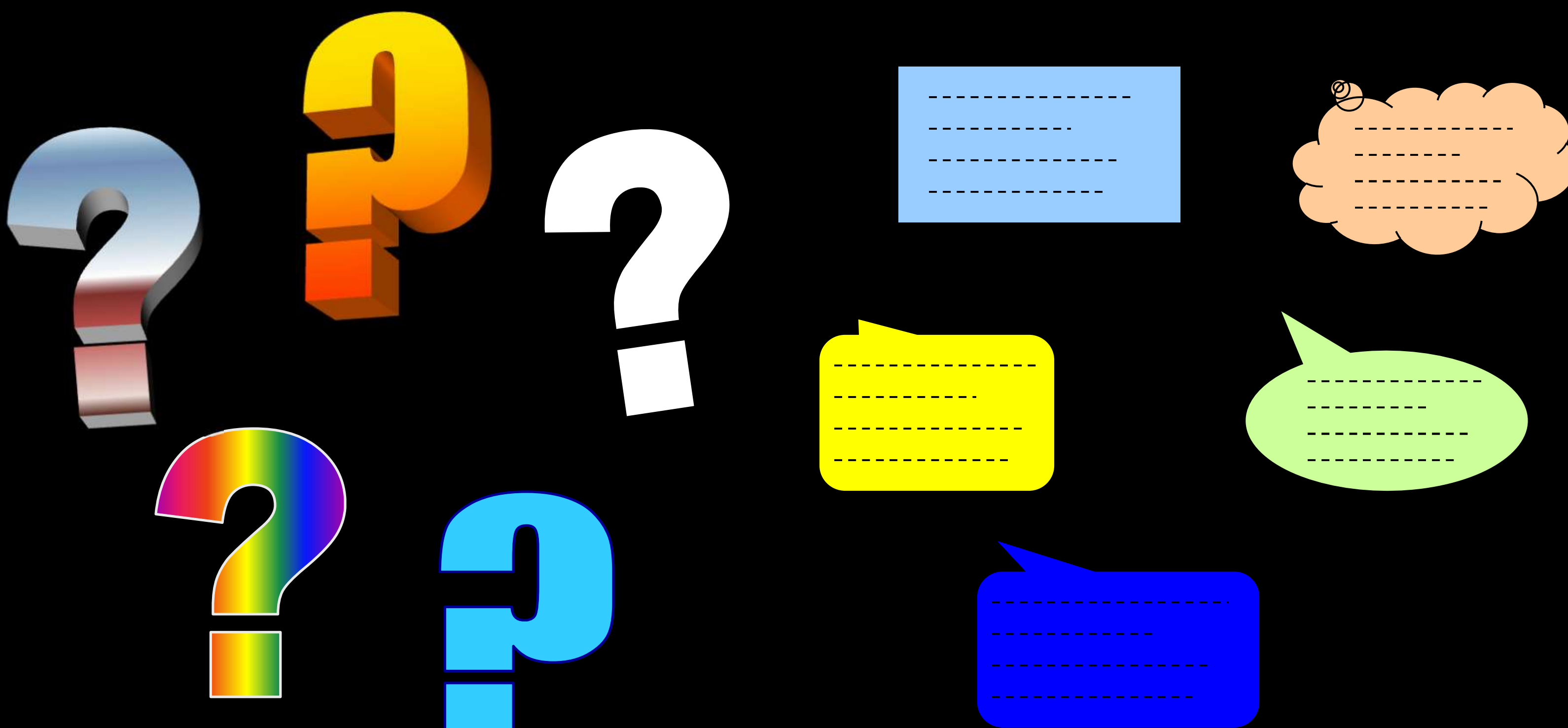
- Introduced RNS with continuous residues
  - Distinct from ordinary RNS
  - Advantages (similar to other hybrid schemes)
- Studied range, accuracy, and tradeoffs
  - Tight bounds for dynamic range
  - Optimal choice of moduli
- Showed link to computational neuroscience
  - Rat's sense of location, navigation
  - Moduli in nature: evolutionary implications

Ongoing and Future Work

- Refine and extend the theoretical framework
  - Arithmetic and algorithmic implications
  - Exact dynamic range, or even tighter bounds
- Study development and application aspects
  - Circuit realization and building blocks
  - Latency, area, and energy implications
- Pursue links with other hybrid D/A methods
  - Mixed implementations?

Thank You for Your Attention

parhami@ece.ucsb.edu
http://www.ece.ucsb.edu/~parhami/



Neurophysiological Discoveries of the 2014 Nobel Prize Winners in Medicine from a Computer Arithmetic Perspective

Behrooz Parhami

Dept. Electrical & Computer Engr.
Univ. of California, Santa Barbara, USA

`parhami@ece.ucsb.edu`

Back-up Slides

RNS Dynamic Range

Product M of the k pairwise relatively prime moduli is the *dynamic range*

$M = m_{k-1} \times \ldots \times m_1 \times m_0$

For RNS(8 | 7 | 5 | 3), $M = 8 \times 7 \times 5 \times 3 = 840$

We can take the range of RNS(8|7|5|3) to be [–420, 419] or any other set of 840 consecutive integers

Negative numbers: Complement relative to M

$\langle -x \rangle_{m_i} = \langle M - x \rangle_{m_i}$

$21 = (5 \mid 0 \mid 1 \mid 0)_{RNS}$

$-21 = (8 - 5 \mid 0 \mid 5 - 1 \mid 0)_{RNS} = (3 \mid 0 \mid 4 \mid 0)_{RNS}$

Here are some example numbers in our default RNS(8 | 7 | 5 | 3):

$(0 \mid 0 \mid 0 \mid 0)_{RNS}$	Represents 0 or 840 or . . .
$(1 \mid 1 \mid 1 \mid 1)_{RNS}$	Represents 1 or 841 or . . .
$(2 \mid 2 \mid 2 \mid 2)_{RNS}$	Represents 2 or 842 or . . .
$(0 \mid 1 \mid 3 \mid 2)_{RNS}$	Represents 8 or 848 or . . .
$(5 \mid 0 \mid 1 \mid 0)_{RNS}$	Represents 21 or 861 or . . .
$(0 \mid 1 \mid 4 \mid 1)_{RNS}$	Represents 64 or 904 or . . .
$(2 \mid 0 \mid 0 \mid 2)_{RNS}$	Represents –70 or 770 or . . .
$(7 \mid 6 \mid 4 \mid 2)_{RNS}$	Represents –1 or 839 or . . .

RNS Encoding and Arithmetic Operations

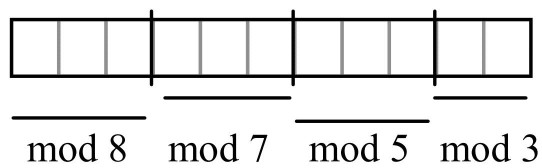


Fig. 4.1 Binary-coded format for RNS(8 | 7 | 5 | 3).

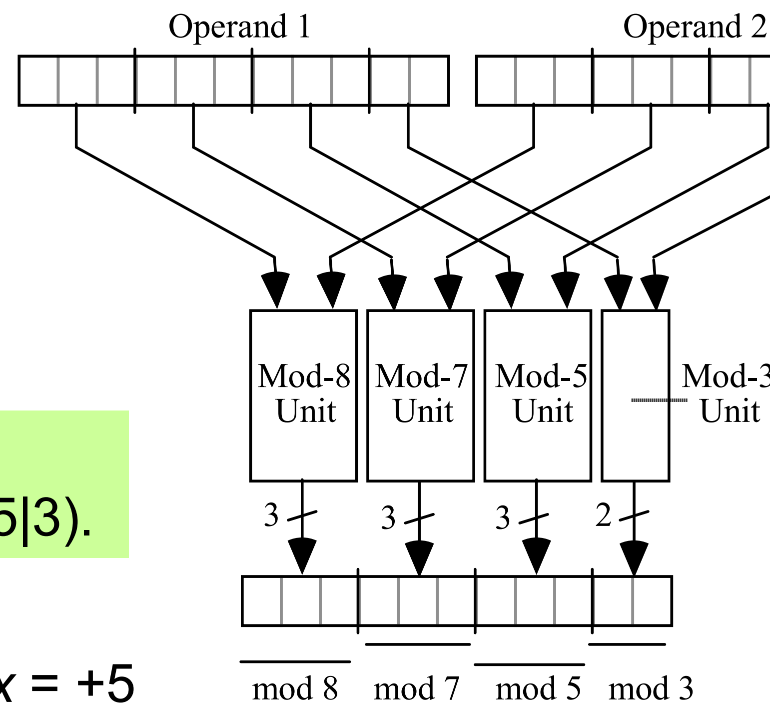


Fig. 4.2 The structure of an adder, subtractor, or multiplier for RNS(8|7|5|3).

Arithmetic in RNS(8 | 7 | 5 | 3)

$(5 \mid 5 \mid 0 \mid 2)_{RNS}$ Represents $x = +5$

$(7 \mid 6 \mid 4 \mid 2)_{RNS}$ Represents $y = -1$

$(4 \mid 4 \mid 4 \mid 1)_{RNS}$ $x + y$: $\langle 5 + 7 \rangle_8 = 4$, $\langle 5 + 6 \rangle_7 = 4$, etc.

$(6 \mid 6 \mid 1 \mid 0)_{RNS}$ $x - y$: $\langle 5 - 7 \rangle_8 = 6$, $\langle 5 - 6 \rangle_7 = 6$, etc.
(alternatively, find $-y$ and add to x)

$(3 \mid 2 \mid 0 \mid 1)_{RNS}$ $x \times y$: $\langle 5 \times 7 \rangle_8 = 3$, $\langle 5 \times 6 \rangle_7 = 2$, etc.

Difficult RNS Arithmetic Operations

Sign test and magnitude comparison are difficult

Example: Of the following RNS(8 | 7 | 5 | 3) numbers:

Which, if any, are negative?
Which is the largest?
Which is the smallest?

Assume a range of [–420, 419]

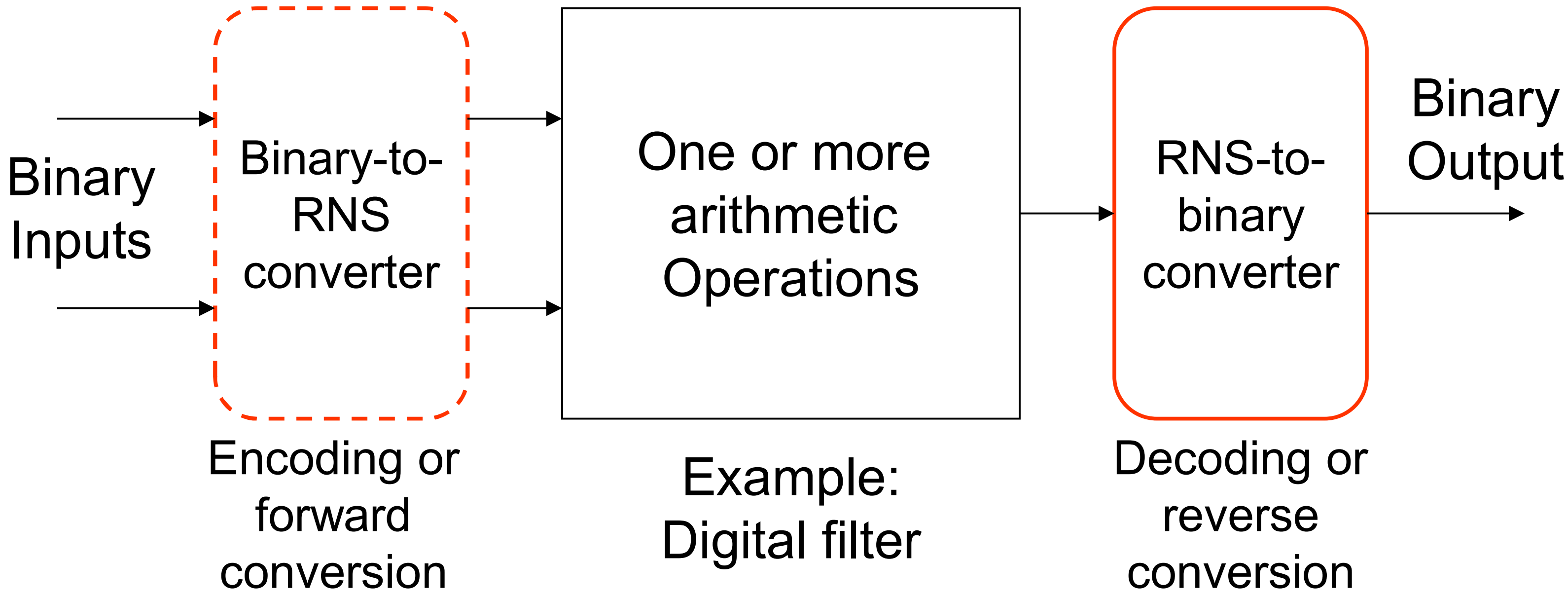
$a = (0 \mid 1 \mid 3 \mid 2)_{RNS}$
$b = (0 \mid 1 \mid 4 \mid 1)_{RNS}$
$c = (0 \mid 6 \mid 2 \mid 1)_{RNS}$
$d = (2 \mid 0 \mid 0 \mid 2)_{RNS}$
$e = (5 \mid 0 \mid 1 \mid 0)_{RNS}$
$f = (7 \mid 6 \mid 4 \mid 2)_{RNS}$

Answers:

$d < c < f < a < e < b$

$-70 < -8 < -1 < 8 < 21 < 64$

Forward and Reverse Conversions



The more the amount of computation performed between the initial forward conversion and final reverse conversion (reconversion), the greater the benefits of RNS representation.

Intuitive Justification for CRT

Puzzle: What number has the remainders of 2, 3, and 2 when divided by the numbers 7, 5, and 3, respectively?

$$x = (2 \mid 3 \mid 2)_{\text{RNS}(7|5|3)} = (?)_{\text{ten}}$$

$(1 \mid 0 \mid 0)_{\text{RNS}(7|5|3)}$ = multiple of 15 that is 1 mod 7 = 15

$(0 \mid 1 \mid 0)_{\text{RNS}(7|5|3)}$ = multiple of 21 that is 1 mod 5 = 21

$(0 \mid 0 \mid 1)_{\text{RNS}(7|5|3)}$ = multiple of 35 that is 1 mod 3 = 70

$$\begin{aligned}(2 \mid 3 \mid 2)_{\text{RNS}(7|5|3)} &= (2 \mid 0 \mid 0) + (0 \mid 3 \mid 0) + (0 \mid 0 \mid 2) \\ &= 2 \times (1 \mid 0 \mid 0) + 3 \times (0 \mid 1 \mid 0) + 2 \times (0 \mid 0 \mid 1) \\ &= 2 \times 15 + 3 \times 21 + 2 \times 70 \\ &= 30 + 63 + 140 \\ &= 233 = 23 \bmod 105\end{aligned}$$

Therefore, $x = (23)_{\text{ten}}$

Example RNS with Special Moduli

For RNS(17 | 16 | 15), the weights of the 3 positions are:

2160 3825 2176

Example: $(x_2, x_1, x_0) = (2 \mid 3 \mid 4)_{RNS}$ represents the number

$$\langle 2160 \times 2 + 3825 \times 3 + 2176 \times 4 \rangle_{4080} = \langle 24{,}499 \rangle_{4080} = 19$$

$2160 = 2^4 \times (2^4 - 1) \times (2^3 + 1) = 2^{11} + 2^7 - 2^4$

$3825 = (2^8 - 1) \times (2^4 - 1) = 2^{12} - 2^8 - 2^4 + 1$

$2176 = 2^7 \times (2^4 + 1) = 2^{11} + 2^7$

$4080 = 2^{12} - 2^4$; thus, to subtract 4080, ignore bit 12 and add 2^4

Reverse converter: Multioperand adder, with shifted x_is as inputs

Limits of Fast Arithmetic in RNS

Known results from number theory

Theorem 4.2: The ith prime p_i is asymptotically $i \ln i$

Theorem 4.3: The number of primes in [1, n] is asymptotically $n / \ln n$

Theorem 4.4: The product of all primes in [1, n] is asymptotically e^n

Implications to speed of arithmetic in RNS

Theorem 4.5: It is possible to represent all k-bit binary numbers in RNS with $O(k / \log k)$ moduli such that the largest modulus has $O(\log k)$ bits

That is, with fast log-time adders, addition needs $O(\log \log k)$ time