Increasing the Information Density of Storage Systems Using the Precision-Resolution Paradigm

Moshe Schwartz, *Member, IEEE,* and Jehoshua Bruck, *Fellow, IEEE*

***Abstract*— Arguably, the most prominent constrained system in storage applications is the (d,k)-RLL (Run-Length Limited) system, where every binary sequence obeys the constraint that every two adjacent 1's are separated by at least d consecutive 0's and at most k consecutive 0's, namely, runs of 0's are length limited. The motivation for the RLL constraint arises mainly from the physical limitations of the read and write technologies in magnetic and optical storage systems.**

We revisit the rationale for the RLL system and reevaluate its relationship to the physical media. As a result, we introduce a new paradigm that better matches the physical constraints. We call the new paradigm the Precision-Resolution (PR) system, where the *write* operation is limited by *precision* and the *read* operation is limited by *resolution*.

We compute the capacity of a general PR system and demonstrate that it provides a significant increase in the information density compared to the traditional RLL system (for identical physical limitations). For example, the capacity of the $(2,10)$-RLL used in CD-ROMs and DVDs is approximately 0.5418, while our PR system provides the capacity of about 0.7725, resulting in a potential increase of about 40% in information density.

***Index Terms*— run-length limited, constrained coding, capacity of constrained channels.**

I. Introduction

THE (d,k)-RLL system is perhaps the most commonly used constraint, appearing in various contexts and applications, including (but not restricted to) magnetic and optical storage, holographic storage, and wireless and fiber-optic communication. It is however easy to forget that RLL coding is but one possible solution to a set of restrictions imposed by the medium, by current sensing and processing technology, or by our own choice. To better understand any extension, we should first describe these restrictions.

In the context of storage applications, for example, one wants to write a stream of bits. Due to the nature of recording devices, when reading back the stream, we receive a series of analog peaks which correspond to the $\mathbf{1}$'s in the original bit stream. The first restriction imposed by such a system is a function of the sensors used for the reading which may cause adjacent peaks to merge if these are positioned too close to each other. In the RLL constraint, this restriction is addressed by requiring that in the bit stream we write, every two adjacent $\mathbf{1}$'s are separated by at least $d \geqslant 0$ consecutive $\mathbf{0}$'s.

On the other hand, the decoding mechanism attempts to acquire the correct number of $\mathbf{0}$'s in a run by measuring the time between two peaks, and dividing it by the duration of a single $\mathbf{0}$. Since no two clocks are identical, the difference in their frequencies causes a drift. If this drift is bounded by $0 < \delta < 1$, then a written run of $\mathbf{0}$'s of duration $t \in R$, may actually end up being decoded as a run of duration $(1-\delta)t < t' < (1+\delta)t$. Obviously, the longer the run of $\mathbf{0}$'s, the more spurious or missing $\mathbf{0}$'s may occur in the decoding. To avoid such a problem, the RLL-constraint chooses to limit the maximum length of any run of $\mathbf{0}$'s to k, where $k \in \mathbb{N}$ is the largest integer for which $(1+\delta)k \leqslant (1-\delta)(k+1)$. We note that since the $\mathbf{1}$'s are used only to punctuate the stream, and we measure the time duration between peaks, the duration of the run includes the duration of the consecutive $\mathbf{0}$'s and the single following $\mathbf{1}$. Thus, for example, any value of $\delta \in (\frac{1}{23}, \frac{1}{21}]$ results in $k = 10$ found in the $(2,10)$-RLL of CD-ROMs and DVDs.

Hence, in order to get a correct decoder's **resolution**, we need to make sure the corresponding (d,k) constraints are satisfied. However, current RLL systems include an unnecessary implicit restriction: The requirement that the set of valid intervals form a sequence of consecutive integers. Having similar motivation, Funk [2] devised an improvement to the RLL scheme called RLL-MS, by replacing this requirement with another equally arbitrary requirement that the set of valid intervals form an arithmetic progression. The interval-modulation scheme introduced by Mukhtar and Bruck [8], [9] generalized RLL to take advantage of higher precision clocks, however, it mainly addressed the encoder/decoder constructions.

Parting ways with the restriction that the time between adjacent peaks be an integer and parameterizing this **precision** constraint leads us to define a sequence of ever more precise systems. These systems are able to measure small discrete time intervals, and in the theoretical limit, reach a system with infinite-precision measurement. The key benefit in our new paradigm is the *separation* (see Figure 1) between the encoder constraints (which relate to the *precision* of writing information into the physical media) and the decoder constraints (which relate to the ability to distinguish between two different signals received by reading the physical media).

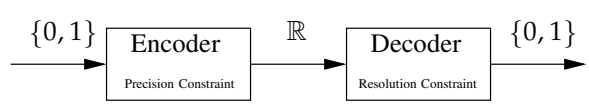

Fig. 1. The precision-resolution framework

We therefore introduce the (p,α,θ)-*PR (precision-resolution) framework*. In this framework (see Figure 1), an *encoder* receives a stream of binary symbols, transforms them into a stream of real numbers which are sent over a noiseless

Moshe Schwartz is with the Department of Electrical Engineering, California Institute of Technology, 1200 E California Blvd., Mail Code 136-93, Pasadena, CA 91125, U.S.A. (e-mail: moosh@paradise.caltech.edu).

Jehoshua Bruck is with the Department of Electrical Engineering, California Institute of Technology, 1200 E California Blvd., Mail Code 136-93, Pasadena, CA 91125, U.S.A. (e-mail: bruck@paradise.caltech.edu).

This work was supported in part by the Caltech Lee Center for Advanced Networking and by NSF grant ANI-0322475.

channel to a *decoder*, which then converts them back to the original binary stream. However, both sides suffer from further constraints:

Encoder Constraint — The encoder has constrained precision. Each transmitted symbol $t \in \mathbb{Q}$ is some $t = m/p$ where $m, p \in \mathbb{N}$, m any integer, and p a fixed integer. We call p the *precision* of the encoder. We denote infinite precision by $p = \infty$, i.e., the framework in which the encoder can send any $t \in \mathbb{R}$.

Decoder Constraints — The decoder suffers from resolution constraints. We may define several different sets of constraints which correspond to different applications. In this paper we consider a set which is motivated mainly (but not only) by storage applications.

- The decoder cannot correctly recover any $t < \alpha$, where $\alpha \in \mathbb{R}$, $\alpha > 0$, is called the *minimum resolution*.
- If $t \in \mathbb{R}$ was transmitted, then the received $t' \in \mathbb{R}$ obeys $(1-\delta)t < t' < (1+\delta)t$, where $\delta \in \mathbb{R}$, $0 < \delta < 1$, is another parameter of the framework. For convenience, we define $\theta = \frac{1+\delta}{1-\delta}$ and call it the *resolution factor*.

We note that the RLL constraint is but one solution to a precision-resolution framework. In the following sections we will present the *precision-resolution constraint* which has a higher capacity than its RLL and RLL-MS counterparts.

In his seminal work, Shannon [10] defined the capacity of a constrained system S as

$$\mathsf{cap}(S) \stackrel{\text{def}}{=} \lim_{n\to\infty} \frac{\log_2 |S(n)|}{n},$$

where $S(n)$, $n \in \mathbb{N}$, denotes the set of strings of length n in S. Further advances in the subject include the work of Adler, Coppersmith, and Hassner [1], who applied symbolic dynamics [5] to the analysis and construction of sliding-block encoders for constrained systems. More extensions may be found in works such as Heegard, Marcus, and Siegel [3], Marcus, Siegel, and Wolf [6], and a review in Marcus, Roth, and Siegel [7].

Since the definition of the RLL constraint in [4], [11], several variants to the RLL constraint were suggested. However, the basic premise that we may use ever growing runs of **0**'s until we reach a run length for which the clock drift may produce a spurious **0**, was never changed. This paradigm was shifted in the work of Mukhtar and Bruck [8], [9] which described an extension to the RLL constraint and gave a variable-bit-rate to variable-bit-rate encoder/decoder scheme for it. In Section II, we describe this extension in the precision-resolution framework, discuss the integral-precision constraint, $(1,\alpha,\theta)$-PR, and calculate its capacity. In Section III we consider the sequence of constraints (p,α,θ)-PR, where $p = 1, 2, \ldots$, and show how it gives a significant improvement to the capacity while the RLL constraint fails. We then find the capacity of the infinite-precision constraint, (∞,α,θ)-PR, and show that it is actually the limit as $p \to \infty$ of the capacity of (p,α,θ)-PR. As a result, we can compare, for example, the capacity of $(2,10)$-RLL used in CD-ROMs and DVDs and is approximately 0.5418, with the capacity of the corresponding $(\infty,3,1.1)$-PR which is approximately 0.7725, thus giving the hope for an increase in capacity of more than 40% in this case. We discuss generalizations to the PR scheme with higher alphabets as well as different resolution constraints in Section IV, and show that under a reasonable set of assumptions, our greedy interval packing is optimal. We summarize our results in Section V.

II. The $(1,\alpha,\theta)$-PR Constraint

The (d,k)-RLL constrained system is, equivalently, the set of all possible strings which are formed by concatenating strings from the set

$$\mathcal{S}_{d,k} \stackrel{\text{def}}{=} \left\{0^d1, 0^{d+1}1, 0^{d+2}1, \ldots, 0^k1\right\}.$$

Because of this form of representation, it was shown in [10] that the capacity of the (d,k)-RLL constraint is $\log_2 r_{d,k}$, where $r_{d,k} \in \mathbb{R}$ is the unique positive root of the equation

$$x^{-(d+1)} + x^{-(d+2)} + x^{-(d+3)} + \cdots + x^{-(k+1)} = 1.$$

We may also think of the strings in $\mathcal{S}_{d,k}$ as a set of symbols which differ only in their duration. For (d,k)-RLL this set is

$$\hat{\mathcal{I}}_{d,k} \stackrel{\text{def}}{=} \{d+1, d+2, \ldots, k+1\}.$$

For the decoder of a (d,k)-RLL stream there is no ambiguity. Any time duration read t', is some $(1-\delta)t < t' < (1+\delta)t$, where $t \in \hat{\mathcal{I}}_{d,k}$ is the written duration and $0 < \delta < 1$ is the clock drift. But since k was chosen such that $(1+\delta)k \leqslant (1-\delta)(k+1)$, for any two distinct elements $t_1, t_2 \in \hat{\mathcal{I}}_{d,k}$ there is no intersection in their δ-neighborhoods: $((1-\delta)t_1, (1+\delta)t_1)$ and $((1-\delta)t_2, (1+\delta)t_2)$.

Essentially, the (d,k)-RLL constraint starts with the minimal time interval allowed by the decoder, $d+1$, and continues to add successive lengths $d+2, d+3, \ldots$ as long as their δ-neighborhoods are disjoint. It stops at the first length $k+1$ whose δ-neighborhood intersects that of $k+2$.

An obvious question asked by Mukhtar and Bruck [8], is why restrict ourselves to successive time durations? We may skip $k+2$ but still be able to use $k+3$ or some other longer length. In [8] it is assumed that some digital clock is governing the system, thus restricting all time durations measured to be integral multiples of the clock period. This brings us to define the integral precision-resolution constrained system.

Definition 1. *Let $\alpha \in \mathbb{R}$, $\alpha > 0$, be the minimum resolution, and let $\theta = \frac{1+\delta}{1-\delta} > 1$ be the resolution factor, where $\delta \in \mathbb{R}$, $0 < \delta < 1$. Then the $(1,\alpha,\theta)$-PR constraint is the set of all streams with symbols from the set*

$$\hat{\mathcal{I}}_{\alpha,\theta} \stackrel{\text{def}}{=} \left\{\lceil \alpha,\theta \rceil^i \mid i = 0,1,2,\ldots\right\} \subset \mathbb{N}$$

where we define

$$\lceil \alpha,\theta \rceil^i \stackrel{\text{def}}{=} \lceil \ldots \lceil \lceil \lceil \alpha \rceil \overbrace{\theta \rceil \theta \rceil \ldots \theta}^{i} \rceil.$$

Example 2. *For $\alpha = 2$ and $\theta = 1.5$, we have*

$$\hat{\mathcal{I}}_{\alpha,\theta} = \{2,3,5,8,12,18,27,\ldots\}.$$

See Figure 2 for an illustration of the disjoint δ-neighborhoods.

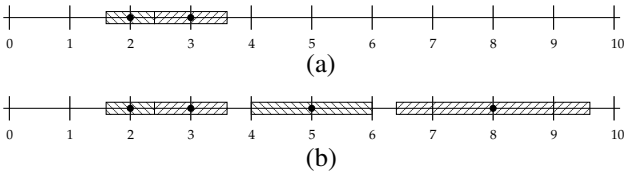

Fig. 2. With $\alpha = 2$ and $\theta = 1.5$ (i.e., $\delta = 0.2$) we see the disjoint δ-neighborhoods of (a) the symbols of the resulting $(1,2)$-RLL, and (b) the first four symbols of $(1,2,1.5)$-PR.

Obviously, the $(1,\alpha,\theta)$-PR constraint obeys the $(1,\alpha,\theta)$-PR framework since all symbols are integers, the smallest one is α, and

$$\lceil \alpha,\theta \rceil^{i+1} = \left\lceil \theta \lceil \alpha,\theta \rceil^{i} \right\rceil \geqslant \theta \lceil \alpha,\theta \rceil^{i}$$

ensures that the δ-neighborhoods of elements in $\hat{\mathcal{I}}_{\alpha,\theta}$ are disjoint. For a practical implementation we may want to limit the size of elements in $\hat{\mathcal{I}}_{\alpha,\theta}$ and so we define

$$\hat{\mathcal{I}}^{m}_{\alpha,\theta} \stackrel{\text{def}}{=} \left\{ \tau \in \hat{\mathcal{I}}_{\alpha,\theta} \mid \tau \leqslant m \right\}.$$

Furthermore, if we define

$$k_\theta = \max\left\{ k \in \mathbb{Z} \mid \theta(k-1) \leqslant k \right\},$$

then $\hat{\mathcal{I}}^{k_\theta}_{\alpha,\theta}$ is exactly the set used in $(\lceil \alpha \rceil - 1, k_\theta - 1)$-RLL.

Theorem 3. *Let $\hat{S}^{m}_{\alpha,\theta}$ denote the constraint with intervals from $\hat{\mathcal{I}}^{m}_{\alpha,\theta}$. Then for every $m < m'$ such that $\left|\hat{\mathcal{I}}^{m}_{\alpha,\theta}\right| < \left|\hat{\mathcal{I}}^{m'}_{\alpha,\theta}\right|$,*

$$\mathsf{cap}(\hat{S}^{m}_{\alpha,\theta}) < \mathsf{cap}(\hat{S}^{m'}_{\alpha,\theta}) < 1.$$

Proof: As was pointed out before, $\mathsf{cap}(\hat{S}^{m}_{\alpha,\theta}) = \log_2 \hat{r}_m$ where $\hat{r}_m$ is the unique positive solution to the equation $\hat{f}_m(x) = 1$, and

$$\hat{f}_m(x) \stackrel{\text{def}}{=} \sum_{\tau \in \hat{\mathcal{I}}^{m}_{\alpha,\theta}} x^{-\tau}.$$

Similarly we define $\hat{r}_{m'}$ and $\hat{f}_{m'}(x)$, where the latter is easily seen to be monotone decreasing and continuous in $(0,\infty)$. Since $\hat{f}_{m'}(2) < 1$ we immediately get that $\mathsf{cap}(\hat{S}^{m'}_{\alpha,\theta}) < 1$. Furthermore,

$$\hat{f}_{m'}(\hat{r}_m) = \sum_{\tau \in \hat{\mathcal{I}}^{m'}_{\alpha,\theta}} \hat{r}_m^{-\tau} = 1 + \sum_{\tau \in \hat{\mathcal{I}}^{m'}_{\alpha,\theta} \setminus \hat{\mathcal{I}}^{m}_{\alpha,\theta}} \hat{r}_m^{-\tau} > 1,$$

and so $\mathsf{cap}(\hat{S}^{m}_{\alpha,\theta}) < \mathsf{cap}(\hat{S}^{m'}_{\alpha,\theta})$. ■

We therefore conclude that adding more symbols to the constraint, strictly increases the capacity. If we consider the infinite set $\hat{\mathcal{I}}_{\alpha,\theta}$, we get the following theorem.

Theorem 4. *Let $\hat{S}_{\alpha,\theta}$ denote the $(1,\alpha,\theta)$-PR constrained system. Then*

$$\mathsf{cap}(\hat{S}_{\alpha,\theta}) = \log_2 \hat{r}_{\alpha,\theta} < 1$$

where $\hat{r}_{\alpha,\theta}$ is the unique solution in $(1,\infty)$ to the equation $\hat{f}_{\alpha,\theta}(x) = 1$ and

$$\hat{f}_{\alpha,\theta}(x) \stackrel{\text{def}}{=} \sum_{i \geqslant 0} x^{-\lceil \alpha,\theta \rceil^{i}}.$$

Proof: We omit the proof that $\mathsf{cap}(\hat{S}_{\alpha,\theta}) = \log_2 \hat{r}_{\alpha,\theta}$ since this is essentially the same as the proof of Theorem 11. From that proof we also get that $\hat{f}_{\alpha,\theta}(x)$ is continuous and monotone decreasing in $(1,\infty)$. We now note the obvious fact that all the exponents in $\hat{f}_{\alpha,\theta}(x)$ are positive integers, and since $\theta > 1$, not all positive integers appear as exponents. Thus, $\hat{f}_{\alpha,\theta}(2) < 1$ and so $\mathsf{cap}(\hat{S}_{\alpha,\theta}) = \log_2 \hat{r}_{\alpha,\theta} < 1$. ■

We see in Figure 3, for $1 < \theta \leqslant 2$, a comparison of the resulting $(0,k_\theta - 1)$-RLL constraint with its extension, the $(1,1,\theta)$-PR constraint. To plot this graph, an approximation was needed since no simple closed form is known for $\hat{f}_{\alpha,\theta}(x)$. Specifically, we added only the first five monomials beyond those of the corresponding RLL constraint.

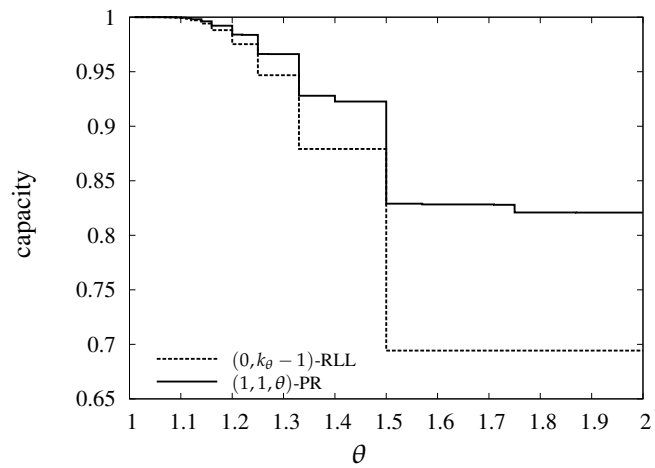

Fig. 3. The capacity of $(0,k_\theta - 1)$-RLL vs. the capacity of $(1,1,\theta)$-PR

III. The General (p,α,θ)-PR Constraint

It is intuitively clear that the rounding up of the symbols of the $(1,\alpha,\theta)$-PR constraint, incurs a penalty in capacity, especially for small values of θ. So while the δ-neighborhoods of $\lceil \alpha \rceil$ and $\lceil \lceil \alpha \rceil \theta \rceil$ are disjoint, the unused gap between them might be fairly large.

All of this is caused by the fact that we restricted ourselves to symbols which are integers. But suppose we are given a system which is $p \in \mathbb{N}$ times more precise[1], i.e., it is able to transmit symbols which are multiples of $1/p$. So now, instead of rounding up the symbols of our constrained system to the nearest integer, we can round them up to the nearest multiple of $1/p$.

Definition 5. *Let $p \in \mathbb{N}$ be the precision, $\alpha \in \mathbb{R}$, $\alpha > 0$, be the minimum resolution, and let $\theta = \frac{1+\delta}{1-\delta} > 1$ be the resolution factor, where $\delta \in \mathbb{R}$, $0 < \delta < 1$. Then the (p,α,θ)-PR constraint is the set of all streams with symbols from the set*

$$\hat{\mathcal{I}}_{p,\alpha,\theta} \stackrel{\text{def}}{=} \left\{ \frac{\lceil p\alpha,\theta \rceil^{i}}{p} \;\middle|\; i = 0,1,2,\ldots \right\}.$$

[1]In practical terms, being p times more precise means the encoder has a clock with p times the original frequency, which allows it to align the peaks to multiples of $1/p$. This does not mean that the encoder writes more densely since the decoder's minimum resolution does not depend on the precision p.

Example 6. *For $p = 2$, $\alpha = 2$ and $\theta = 1.5$, we have*

$$\hat{\mathcal{I}}_{p,\alpha,\theta} = \{2, 3, 4.5, 7, 10.5, 16, 24, \dots\}.$$

See Figure 4 for an illustration of the disjoint δ-neighborhoods which shows the shrinking of the gap between adjacent neighborhoods as the precision grows.

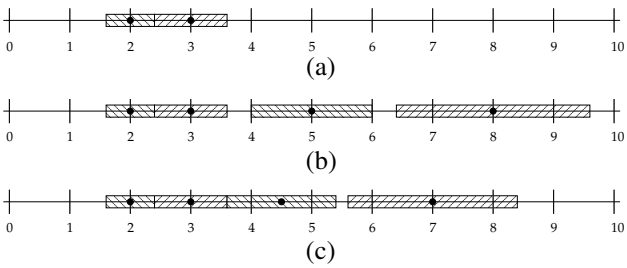

Fig. 4. With $\alpha = 2$ and $\theta = 1.5$ (i.e., $\delta = 0.2$) we see the disjoint δ-neighborhoods of (a) the symbols of the resulting $(1, 2)$-RLL, (b) the first four symbols of $(1, 2, 1.5)$-PR, and (c) the first four symbols of $(2, 2, 1.5)$-PR.

The following theorem shows that we never lose capacity when using a constraint which is p times more precise.

Theorem 7. *Let $p \in \mathbb{N}$, $\alpha, \theta \in \mathbb{R}$, $\alpha > 0$, $\theta > 1$. Then*

$$\mathsf{cap}(\hat{S}_{1,\alpha,\theta}) \leqslant \mathsf{cap}(\hat{S}_{p,\alpha,\theta}) = p \cdot \mathsf{cap}(\hat{S}_{1,p\alpha,\theta})$$

which holds in equality iff $p \lceil \alpha, \theta \rceil^i = \lceil p\alpha, \theta \rceil^i$ for all $i \geqslant 0$.

Proof: First, since the the symbols of (p, α, θ)-PR are exactly the symbols of $(1, p\alpha, \theta)$-PR, only scaled down by a factor of p, there is a 1-1 correspondence between strings of length[2] n in (p, α, θ)-PR, and strings of length pn in $(1, p\alpha, \theta)$-PR. Thus,

$$\mathsf{cap}(\hat{S}_{p,\alpha,\theta}) = p \cdot \mathsf{cap}(\hat{S}_{1,p\alpha,\theta}).$$

Let us now define

$$\hat{f} = \sum_{i \geqslant 0} x^{-\lceil \alpha,\theta \rceil^i} \qquad \text{and} \qquad \hat{f}_p = \sum_{i \geqslant 0} x^{-\lceil p\alpha,\theta \rceil^i}$$

so that $\mathsf{cap}(\hat{S}_{1,\alpha,\theta}) = \log_2 \hat{r}$ and $\mathsf{cap}(\hat{S}_{1,p\alpha,\theta}) = \log_2 \hat{r}_p$, where $\hat{r}, \hat{r}_p \in \mathbb{R}$ are the unique positive solutions to $\hat{f}(x) = 1$ and $\hat{f}_p(x) = 1$ respectively.

We note that $\hat{f}(x^p) \leqslant \hat{f}_p(x)$ for all $x > 1$. This is because $p \in \mathbb{N}$, so for any $\beta \in \mathbb{R}$, $\beta > 0$, we get $p \lceil \beta \rceil \geqslant \lceil p\beta \rceil$ and thus $x^{-p\lceil \alpha,\theta \rceil^i} \leqslant x^{-\lceil p\alpha,\theta \rceil^i}$. Thus, $\hat{f}(r_p^p) \leqslant \hat{f}_p(r_p) = 1$, and since $\hat{f}(x)$ is continuous and monotone decreasing in $(1, \infty)$, this means that

$$\mathsf{cap}(\hat{S}_{1,\alpha,\theta}) \leqslant p \cdot \mathsf{cap}(\hat{S}_{1,p\alpha,\theta}).$$

Finally, since $x^{-p\lceil \alpha,\theta \rceil^i} \leqslant x^{-\lceil p\alpha,\theta \rceil^i}$, equality holds in the above iff $p \lceil \alpha, \theta \rceil^i = \lceil p\alpha, \theta \rceil^i$ for all $i \geqslant 0$. ■

We note that the requirement that p be an integer is necessary for Theorem 7. This is because for $\mu, \beta \in \mathbb{R}$, and $\mu, \beta > 0$, we do not know which of $\mu \lceil \beta \rceil$ and $\lceil \mu\beta \rceil$ is greater.

[2]The length of a string is the sum of the lengths of its symbols, where the length of a symbol is simply its value.

Thus, for example, there are values of θ for which we get $\mathsf{cap}(\hat{S}_{2,1,\theta}) > \mathsf{cap}(\hat{S}_{3,1,\theta})$. In Figure 5, we see $\mathsf{cap}(\hat{S}_{p,1,\theta})$ for $p = 1, 2, 4, 8$.

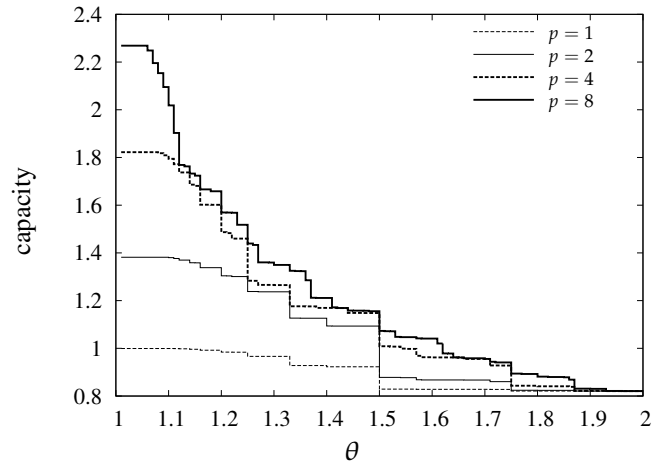

Fig. 5. The capacity of $(p, 1, \theta)$-PR, for $p = 1, 2, 4, 8$.

Returning to the RLL constraint, being p times more precise usually translates into having a digital clock which runs p times faster. While we gain from faster clocks when using (p, α, θ)-PR, the case is often worse in (d, k)-RLL. This is because when using a clock which runs p times faster, we have to use $(p(d+1) - 1, k)$-RLL since k is not affected by the speed of the clock, but the minimum time between adjacent peaks becomes $p(d+1)$ clock ticks. However, we have $p \cdot \mathsf{cap}(S_{p(d+1)-1,k}) = 0$ when $p(d+1) - 1 \geqslant k$, which for most practical values of p, d, and k, is the case.

If we look at Figure 5 there appears to be an upper bound on the capacity as we continue to increase the precision. This is indeed the case as will shall prove shortly. First, we define the appropriate constrained system with infinite-precision.

Definition 8. *Let $\alpha \in \mathbb{R}$, $\alpha > 0$, be the minimum resolution, and let $\theta = \frac{1+\delta}{1-\delta} > 1$ be the resolution factor, where $\delta \in \mathbb{R}$, $0 < \delta < 1$. Then the (∞, α, θ)-PR constraint is the set of all streams with symbols from the set*

$$\mathcal{I}_{\alpha,\theta} \stackrel{\text{def}}{=} \left\{ \alpha\theta^i \mid i = 0, 1, 2, \dots \right\} \subset \mathbb{R}.$$

Example 9. *For $\alpha = 2$ and $\theta = 1.5$, we have*

$$\mathcal{I}_{\alpha,\theta} = \{2, 3, 4.5, 6.75, 10.125, 15.1875, 22.78125, \dots\}.$$

See Figure 6 for an illustration of the disjoint δ-neighborhoods which shows the total elimination of the gaps between adjacent neighborhoods with infinite precision.

Since an (∞, α, θ)-PR system is no longer made up of discrete bits, we need the analog of length n strings. We define the *τ-header*, $\tau \in \mathbb{R}$, of such a stream, as the ordered set of symbols appearing between the beginning of the stream and time τ. If we observe the τ-header, we may see several symbols encoded in it one after the other. The last complete symbol may not necessarily reach time τ exactly. The remaining part

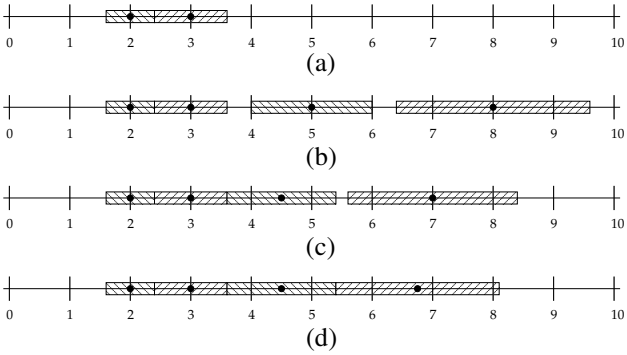

Fig. 6. With $\alpha = 2$ and $\theta = 1.5$ (i.e., $\delta = 0.2$) we see the disjoint δ-neighborhoods of (a) the symbols of the resulting $(1,2)$-RLL, (b) the first four symbols of $(1,2,1.5)$-PR, (c) the first four symbols of $(2,2,1.5)$-PR, and (d) the first four symbols of $(\infty,2,1.5)$-PR.

of the time interval is the beginning of another encoded symbol whose end we do not see, and which we call the *tail*.

Example 10. *For $\alpha = 1$, $\theta = 1.5$ the allowed symbols are of duration $1 = \alpha\theta^0, 1.5 = \alpha\theta^1, 2.25 = \alpha\theta^2, \ldots$, and so on. The following 3-headers may be seen: $(1,1,1)$, $(1.5,1.5)$, $(1,1.5,-)$, $(1.5,1,-)$, $(1,1,-)$, $(2.25,-)$, $(1.5,-)$, $(1,-)$, and $(-)$. The $-$ sign denotes the tail.*

We can now proceed with calculating the capacity of the (∞,α,θ)-PR constraint. For simplicity, we handle the $\alpha = 1$ case first, and then prove the general case.

Theorem 11. *Let S_θ denote the $(\infty,1,\theta)$-PR constrained system, $\theta > 1$. Then*

$$\mathsf{cap}(S_\theta) = \log_2 r_\theta$$

where r_θ is the unique solution in $(1,\infty)$ to the equation $f_\theta(x) = 1$ and

$$f_\theta(x) \stackrel{\text{def}}{=} \sum_{i\geqslant 0} x^{-\theta^i}.$$

Proof: The first thing we do is prove that r_θ actually exists and is unique. For any $x \in (1,\infty)$, the sequence of partial sums $\sigma_j(x) = \sum_{i=0}^{j} x^{-\theta^i}$ is strictly increasing and bounded from above by the geometric sum $\sum_{i=0}^{\infty} x^{-1-i(\theta-1)} = (x - x^{2-\theta})^{-1}$ and thus converges. Now let $x \in [a,b] \subset (1,\infty)$ and then $x^{-\theta^i} \leqslant a^{-\theta^i}$ and since we have already noted that the sequence $\sigma_j(a)$ converges, by the Weierstrass M-test $\sigma_j(x)$ converges uniformly in $[a,b]$ and is therefore continuous. Thus $f_\theta(x)$ is continuous for any $x \in (1,\infty)$ and is easily seen to be strictly decreasing. Finally, since

$$\lim_{x\to 1^+} f_\theta(x) = \infty \qquad \text{and} \qquad \lim_{x\to\infty} f_\theta(x) = 0$$

there exists a unique r_θ as required.

Let us denote by $S_\theta(\tau)$ the set of all τ-headers of S_θ. Obviously,

$$|S_\theta(\tau)| = \begin{cases} 1 + \sum_{i=0}^{\infty} \left|S_\theta(\tau-\theta^i)\right| & \tau \geqslant 0 \\ 0 & \tau < 0 \end{cases}$$

For technical simplicity let us define $\overline{S}_\theta(\tau)$ to be the set of all τ-headers of S_θ with tail strictly smaller than 1. It is easy to see that

$$\left|\overline{S}_\theta(\tau)\right| = \begin{cases} \sum_{i=0}^{\infty} \left|\overline{S}_\theta(\tau-\theta^i)\right| & \tau \geqslant 1 \\ 1 & 0 \leqslant \tau < 1 \\ 0 & \tau < 0 \end{cases}$$

We contend that $|S_\theta(\tau)| \leqslant \lfloor\tau\rfloor \left|\overline{S}_\theta(\tau)\right|$ for $\tau \geqslant 1$. This is because any τ-header of $\overline{S}_\theta(\tau)$ contains at most $\lfloor\tau\rfloor$ complete symbols, and since its tail is strictly shorter than 1, we can create at most $\lfloor\tau\rfloor$ distinct τ-headers in $S_\theta(\tau)$ by repeatedly merging the tail with the last complete symbol. To complete the argument, we can get any τ-header of $S_\theta(\tau)$ that way because we can take any τ-header with tail of length 1 or more, and repeatedly stuff complete symbols of length 1 while reducing the tail by 1, to get a τ-header from $\overline{S}_\theta(\tau)$. See Figure 7 for an example of merging and stuffing.

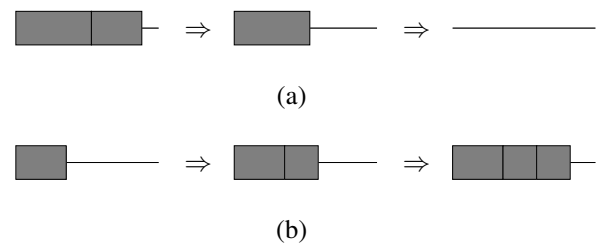

Fig. 7. Two examples with 4.25-headers: (a) Merging $(2.25,1.5,-)$ to $(2.25,-)$ to $(-)$. (b) Stuffing $(1.5,-)$ to $(1.5,1,-)$ to $(1.5,1,1,-)$.

We now prove by induction[3] that $\left|\overline{S}_\theta(\tau)\right| \leqslant r_\theta^\tau$. For the induction base we point out that obviously $\left|\overline{S}_\theta(\tau)\right| = 0 \leqslant r_\theta^\tau$ for all $\tau \in (-\infty,0)$ by definition. Furthermore, since $r_\theta > 1$, $\left|\overline{S}_\theta(\tau)\right| = 1 \leqslant r_\theta^\tau$ for all $\tau \in [0,1)$. For the induction hypothesis, let us assume that $\left|\overline{S}_\theta(\tau)\right| \leqslant r_\theta^\tau$ for all $\tau \in (-\infty,n)$, $n \in \mathbb{N}$. We then prove the claim also holds for $\tau \in [n,n+1)$. Let $\tau \in [n,n+1)$, then

$$\left|\overline{S}_\theta(\tau)\right| = \sum_{i=0}^{\infty} \left|\overline{S}_\theta(\tau-\theta^i)\right| \leqslant \sum_{i=0}^{\infty} r_\theta^{\tau-\theta^i} = r_\theta^\tau \sum_{i=0}^{\infty} r_\theta^{-\theta^i} = r_\theta^\tau$$

where we are able to use the induction hypothesis since $\theta^i \geqslant 1$ and so $\tau - \theta^i \in (-\infty,n)$. It follows that

$$\begin{aligned}\mathsf{cap}(S_\theta) &= \lim_{\tau\to\infty} \frac{\log_2 |S_\theta(\tau)|}{\tau} \leqslant \lim_{\tau\to\infty} \frac{\log_2\left(\lfloor\tau\rfloor \left|\overline{S}_\theta(\tau)\right|\right)}{\tau} \\ &\leqslant \log_2 r_\theta.\end{aligned}$$

We now want to prove that $\mathsf{cap}(S_\theta) \geqslant \log_2 r_\theta$. We contend that $|S_\theta(\tau)| \geqslant \frac{1}{r_\theta} r_\theta^\tau$ for all $\tau \geqslant 0$. We do this again by induction. For the induction base take any $\tau \in [0,1)$ and then

$$|S_\theta(\tau)| = 1 \geqslant \frac{1}{r_\theta} r_\theta^\tau.$$

For the induction hypothesis we assume that $|S_\theta(\tau)| \geqslant \frac{1}{r_\theta} r_\theta^\tau$ for all $\tau \in [0,n)$, $n \in \mathbb{N}$. We prove the claim also holds for

[3]Though the domain of $\left|\overline{S}_\theta(\tau)\right|$ is $\mathbb{R}$, the induction will take place on $\mathbb{N}$.

$\tau \in [n, n+1)$. Taking $\tau \in [n, n+1)$ it follows that,

$$\begin{aligned}
|S_\theta(\tau)| &= 1 + \sum_{i=0}^{\infty} \left| S_\theta(\tau - \theta^i) \right| \\
&= 1 + \sum_{i=0}^{\lfloor \log_\theta \tau \rfloor} \left| S_\theta(\tau - \theta^i) \right| \\
&\geqslant 1 + \sum_{i=0}^{\lfloor \log_\theta \tau \rfloor} \frac{1}{r_\theta} r_\theta^{\tau - \theta^i} \\
&= 1 + \sum_{i=0}^{\infty} \frac{1}{r_\theta} r_\theta^{\tau - \theta^i} - \sum_{i=\lfloor \log_\theta \tau \rfloor + 1}^{\infty} \frac{1}{r_\theta} r_\theta^{\tau - \theta^i} \\
&= \frac{1}{r_\theta} r_\theta^{\tau} + 1 - \sum_{i=\lfloor \log_\theta \tau \rfloor + 1}^{\infty} \frac{1}{r_\theta} r_\theta^{\tau - \theta^i} \\
&\geqslant \frac{1}{r_\theta} r_\theta^{\tau} + 1 - \sum_{i=0}^{\infty} \frac{1}{r_\theta} r_\theta^{1 - \theta^i} \\
&= \frac{1}{r_\theta} r_\theta^{\tau}.
\end{aligned}$$

Thus we get

$$\mathsf{cap}(S_\theta) = \lim_{\tau \to \infty} \frac{\log_2 |S_\theta(\tau)|}{\tau} \geqslant \log_2 r_\theta$$

which completes the proof. ■

To translate the capacity of $(\infty, 1, \theta)$-PR to the capacity of (∞, α, θ)-PR with any $\alpha > 0$, we have the following theorem.

Theorem 12. *Let $S_{\alpha,\theta}$ denote the (∞, α, θ)-PR constrained system, $\alpha > 0$, $\theta > 1$. Then*

$$\mathsf{cap}(S_{\alpha,\theta}) = \frac{1}{\alpha} \cdot \mathsf{cap}(S_{1,\theta}).$$

Proof: Just observe the simple 1-1 correspondence between τ-headers of $S_{1,\theta}$ and $\alpha\tau$-headers of $S_{\alpha,\theta}$ achieved through simple scaling by a factor of α. ■

The capacity of (∞, α, θ)-PR is an upper bound on the capacity of any (p, α, θ)-PR, $p \in \mathbb{N}$, and is actually the limit as $p \to \infty$. This is shown in the next two theorems.

Theorem 13. *For any $p \in \mathbb{N}$, $\alpha, \theta \in \mathbb{R}$, $\alpha > 0$, $\theta > 1$, let $\hat{S}_{p,\alpha,\theta}$ denote the (p, α, θ)-PR constraint, and $S_{\alpha,\theta}$ denote the (∞, α, θ)-PR constraint. Then*

$$\mathsf{cap}(\hat{S}_{p,\alpha,\theta}) \leqslant \mathsf{cap}(S_{\alpha,\theta}).$$

Proof: Since $\lceil p\alpha, \theta \rceil^i / p \geqslant \alpha\theta^i$, take any string of length n from $\hat{S}_{p,\alpha,\theta}$, and replace every symbol of duration $\lceil p\alpha, \theta \rceil^i / p$ with a symbol of duration $\alpha\theta^i$ and by extending the tail, getting an n-header from $S_{\alpha,\theta}$. Thus $\left|\hat{S}_{p,\alpha,\theta}(n)\right| \leqslant |S_{\alpha,\theta}(n)|$ which proves the claim. ■

Theorem 14. *With notation as before,*

$$\lim_{p \to \infty} \mathsf{cap}(\hat{S}_{p,\alpha,\theta}) = \mathsf{cap}(S_{\alpha,\theta}).$$

Proof: Let us define

$$\hat{f}_{p,\alpha,\theta}(x) = \sum_{i \geqslant 0} x^{-\frac{\lceil p\alpha, \theta \rceil^i}{p}} \qquad \text{and} \qquad f_{\alpha,\theta}(x) = \sum_{i \geqslant 0} x^{-\alpha\theta^i},$$

and denote by $\hat{r}_{p,\alpha,\theta}$, $r_{\alpha,\theta}$ the unique solutions in $(1, \infty)$ to $\hat{f}_{p,\alpha,\theta}(x) = 1$ and $f_{\alpha,\theta}(x) = 1$ respectively. Thus, we get $\mathsf{cap}(\hat{S}_{p,\alpha,\theta}) = \log_2 \hat{r}_{p,\alpha,\theta}$ and $\mathsf{cap}(S_{\alpha,\theta}) = \log_2 r_{\alpha,\theta}$.

Since $\sum_{i \geqslant 0} x^{-\lceil p\alpha, \theta \rceil^i / p}$ also passes the Weierstrass M-test conditions for any $[a, b] \subset (1, \infty)$, it follows that,

$$\begin{aligned}
\lim_{p \to \infty} \hat{f}_{p,\alpha,\theta}(x) &= \lim_{p \to \infty} \sum_{i \geqslant 0} x^{-\frac{\lceil p\alpha, \theta \rceil^i}{p}} = \sum_{i \geqslant 0} \lim_{p \to \infty} x^{-\frac{\lceil p\alpha, \theta \rceil^i}{p}} \\
&= \sum_{i \geqslant 0} x^{-\alpha\theta^i} = f_{\alpha,\theta}(x),
\end{aligned}$$

and so $\lim_{p \to \infty} \hat{r}_{p,\alpha,\theta} = r_{\alpha,\theta}$, thus completing the proof. ■

The capacity of the $(\infty, 1, \theta)$-PR constraint is shown in Figure 8. Simple root-finding techniques were used together with limited precision to approximate the roots of $f_\theta(x) = 1$.

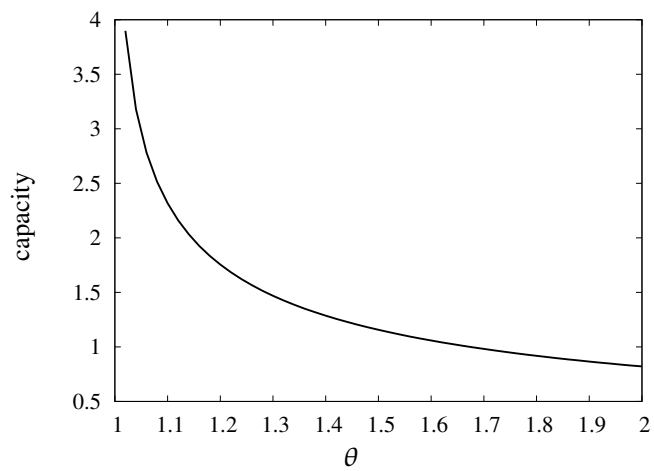

Fig. 8. The capacity of $(\infty, 1, \theta)$-PR

IV. Generalization and Optimality

A. Larger alphabets

In some applications, another degree of freedom is possible in the form of a larger alphabet. In applications such as magnetic recording devices and optical storage, the recording is done using intervals of two "colors", where each two adjacent intervals have different colors and the interval is defined by the color transitions at its beginning and end. By taking the derivative of this stream of intervals we get the traditional representation as a stream of binary symbols where the **1**'s denote the transitions, and the amount of **0**'s between them count the length of the interval.

Some new applications allow intervals to be colored by more than two colors. Though we will not use this representation, again we may take the derivative and represent the encoded stream as a string of symbols over $\mathbb{Z}_q$ where q is the number of colors. Each non-zero position represents a transition by the difference in color values in $\mathbb{Z}_q$, while runs of **0**'s determine the length of the encoded interval.

The precision resolution framework for alphabets of size q, which we denote $(p, \alpha, \theta)_q$-PR, is defined as before, only now the symbols transmitted over the channel come from the

alphabet $\mathbb{R} \times (\mathbb{Z}_q \setminus \{0\})$. The omission of $\mathbf{0}$ from $\mathbb{Z}_q$ comes to ensure that two adjacent intervals do not have the same color, for example by defining that the element from $\mathbb{Z}_q \setminus \{0\}$ encodes the difference in colors between the current symbol and the previous one. Though the actual number of streams should be multiplied by q to account for the choice of color for the first interval (all other colors are then determined by the difference in colors as said before), we will ignore this factor for the sake of simplicity since it does not affect the capacity calculation. Thus, from now on, by abuse of terminology, if $(l, c) \in \mathbb{R} \times (\mathbb{Z}_q \setminus \{0\})$ is a transmitted symbol then we call c the *color* of the symbol.

Since all the theorems generalize in a similar fashion, we will prove only the generalization of Theorem 11.

Theorem 15. *Let $S_{\theta,q}$ denote the $(\infty, 1, \theta)_q$-PR constrained system, $\theta > 1$, $q \geqslant 2$. Then*

$$\mathsf{cap}(S_{\theta,q}) = \log_2 r_{\theta,q}$$

where $r_{\theta,q}$ is the unique solution in $(1, \infty)$ to the equation $f_\theta(x) = \frac{1}{q-1}$ and

$$f_\theta(x) \stackrel{\text{def}}{=} \sum_{i \geqslant 0} x^{-\theta^i}.$$

Proof: We start by noting that $f_\theta(x)$ is exactly the same as in Theorem 11, and so the same arguments may be used to show that indeed there exists a unique $r_{\theta,q}$ as claimed.

Let us denote by $S_{\theta,q}(\tau)$ the set of all τ-headers of $S_{\theta,q}$. Obviously,

$$\left|S_{\theta,q}(\tau)\right| = \begin{cases} q - 1 + \sum_{i=0}^{\infty}(q-1)\left|S_\theta(\tau - \theta^i)\right| & \tau > 0 \\ 1 & \tau = 0 \\ 0 & \tau < 0 \end{cases}$$

For technical simplicity let us define $\overline{S}_{\theta,q}(\tau)$ to be the set of all τ-headers of $S_{\theta,q}$ with tail strictly smaller than 1. It is easy to see that

$$\left|\overline{S}_{\theta,q}(\tau)\right| = \begin{cases} \sum_{i=0}^{\infty}(q-1)\left|\overline{S}_\theta(\tau - \theta^i)\right| & \tau \geqslant 1 \\ q - 1 & 0 < \tau < 1 \\ 1 & \tau = 0 \\ 0 & \tau < 0 \end{cases}$$

We contend that $\left|S_{\theta,q}(\tau)\right| \leqslant \lfloor \tau \rfloor \left|\overline{S}_{\theta,q}(\tau)\right|$ for $\tau \geqslant 1$. This is because any τ-header of $\overline{S}_{\theta,q}(\tau)$ contains at most $\lfloor \tau \rfloor$ complete symbols, and since its tail is strictly shorter than 1, we can create at most $\lfloor \tau \rfloor$ distinct τ-headers in $S_{\theta,q}(\tau)$ by repeatedly merging the tail with the last complete symbol and setting the tail's color to that of the merged symbol . To complete the argument, we can get any τ-header of $S_{\theta,q}(\tau)$ that way because we can take any τ-header with tail of length 1 or more, and repeatedly stuff complete symbols of length 1 and same color as the tail, while reducing the tail by 1, to get a τ-header from $\overline{S}_{\theta,q}(\tau)$.

We now prove by induction that $\left|\overline{S}_{\theta,q}(\tau)\right| \leqslant (q-1) r_{\theta,q}^{\tau}$. For the induction base we point out that obviously $\left|\overline{S}_{\theta,q}(\tau)\right| = 0 \leqslant (q-1) r_{\theta,q}^{\tau}$ for all $\tau \in (-\infty, 0)$ by definition. Furthermore, since $r_{\theta,q} > 1$, $\left|\overline{S}_{\theta,q}(\tau)\right| \leqslant q - 1 \leqslant (q-1) r_{\theta,q}^{\tau}$ for all $\tau \in [0, 1)$. For the induction hypothesis, let us assume that $\left|\overline{S}_{\theta,q}(\tau)\right| \leqslant (q-1) r_{\theta,q}^{\tau}$ for all $\tau \in (-\infty, n)$, $n \in \mathbb{N}$. We then prove the claim also holds for $\tau \in [n, n+1)$. Let $\tau \in [n, n+1)$, then

$$\begin{aligned} \left|\overline{S}_{\theta,q}(\tau)\right| &= (q-1) \sum_{i=0}^{\infty} \left|\overline{S}_{\theta,q}(\tau - \theta^i)\right| \leqslant (q-1)^2 \sum_{i=0}^{\infty} r_{\theta,q}^{\tau - \theta^i} \\ &= (q-1)^2 r_{\theta,q}^{\tau} \sum_{i=0}^{\infty} r_{\theta,q}^{-\theta^i} = (q-1) r_\theta^{\tau} \end{aligned}$$

where we are able to use the induction hypothesis since $\theta^i \geqslant 1$ and so $\tau - \theta^i \in (-\infty, n)$. It follows that

$$\begin{aligned} \mathsf{cap}(S_{\theta,q}) &= \lim_{\tau \to \infty} \frac{\log_2 \left|S_{\theta,q}(\tau)\right|}{\tau} \\ &\leqslant \lim_{\tau \to \infty} \frac{\log_2 \left(\lfloor \tau \rfloor \left|\overline{S}_{\theta,q}(\tau)\right|\right)}{\tau} \leqslant \log_2 r_{\theta,q}. \end{aligned}$$

We now want to prove that $\mathsf{cap}(S_{\theta,q}) \geqslant \log_2 r_{\theta,q}$. We contend that $\left|S_{\theta,q}(\tau)\right| \geqslant \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau}$ for all $\tau \geqslant 0$. We do this again by induction. For the induction base take any $\tau \in [0, 1)$ and then

$$\left|S_{\theta,q}(\tau)\right| \geqslant 1 \geqslant \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau}.$$

For the induction hypothesis we assume that $\left|S_{\theta,q}(\tau)\right| \geqslant \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau}$ for all $\tau \in [0, n)$, $n \in \mathbb{N}$. We prove the claim also holds for $\tau \in [n, n+1)$. Taking $\tau \in [n, n+1)$ it follows that,

$$\begin{aligned} |S_\theta(\tau)| &= q - 1 + (q-1) \sum_{i=0}^{\infty} \left|S_{\theta,q}(\tau - \theta^i)\right| \\ &= (q-1) \left(1 + \sum_{i=0}^{\lfloor \log_\theta \tau \rfloor} \left|S_{\theta,q}(\tau - \theta^i)\right|\right) \\ &\geqslant (q-1) \left(1 + \sum_{i=0}^{\lfloor \log_\theta \tau \rfloor} \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau - \theta^i}\right) \\ &= (q-1) \left(1 + \sum_{i=0}^{\infty} \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau - \theta^i} - \sum_{i = \lfloor \log_\theta \tau \rfloor + 1}^{\infty} \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau - \theta^i}\right) \\ &= (q-1) \left(1 + \frac{r_{\theta,q}^{\tau}}{(q-1) r_{\theta,q}} - \sum_{i = \lfloor \log_\theta \tau \rfloor + 1}^{\infty} \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau - \theta^i}\right) \\ &\geqslant (q-1) \left(1 + \frac{r_{\theta,q}^{\tau}}{(q-1) r_{\theta,q}} - \sum_{i=0}^{\infty} \frac{1}{r_{\theta,q}} r_{\theta,q}^{1 - \theta^i}\right) \\ &= \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau} + q - 2 \geqslant \frac{1}{r_{\theta,q}} r_{\theta,q}^{\tau}. \end{aligned}$$

Thus we get

$$\mathsf{cap}(S_{\theta,q}) = \lim_{\tau \to \infty} \frac{\log_2 \left|S_{\theta,q}(\tau)\right|}{\tau} \geqslant \log_2 r_{\theta,q}$$

which completes the proof. ■

The capacity of the $(\infty, 1, \theta)_q$-PR constraint for $q = 2, 3, 4, 5$ is shown in Figure 9. Simple root-finding techniques were used together with limited precision to approximate the roots of $f_\theta(x) = \frac{1}{q-1}$.

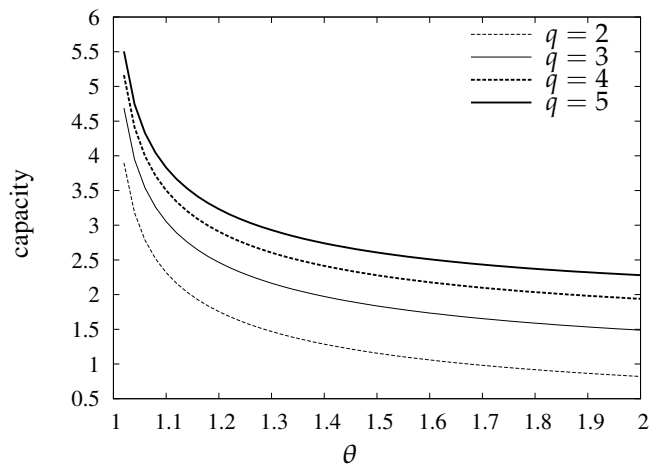

Fig. 9. The capacity of $(\infty, 1, \theta)_q$-PR for $q = 2, 3, 4, 5$

B. General receiver errors and optimality

Up until now we have considered just one type of error at the receiver, a multiplicative error caused by clock drift. However, in many cases the situation is more complex and other types of errors may be involved and even combined together, such as additive jitter error with multiplicative clock-drift error.

In a more general setting, we associate two *resolution-restriction functions*, $L, R : \mathbb{R} \to \mathbb{R}$, with the receiver. When a symbol $t \in \mathbb{R}$ is transmitted, the receiver may read any of the values in the interval $(L(t), R(t))$. For example, a clock-drift error is defined by $L(t) = (1-\delta)t$ and $R(t) = (1+\delta)t$, a jitter error is defined by $L(t) = t - \Delta$ and $R(t) = t + \Delta$, while their combination may be defined by $L(t) = \min\{(1-\delta)t, t-\Delta\}$ and $R(t) = \max\{(1+\delta)t, t+\Delta\}$.

We note that underlying the previous sections is a simple algorithm, which given a precision p, minimum resolution α, and resolution-restriction functions $L(t)$ and $R(t)$, chooses a symbol set greedily. More precisely, we start with an empty symbol set. Each time we add the minimal symbol $t \in \mathbb{R}$ such that it is correctly aligned according to p, is above the minimal resolution α, and $(L(t), R(t))$ has an empty intersection with $(L(t'), R(t'))$ for any t' of the previously selected symbols. We call this algorithm the *precision-resolution symbol-selection algorithm*.

Theorem 16. *Given precision p, minimum resolution α, and resolution-restriction function $L(t)$ and $R(t)$, let $\mathcal{I}$ be the symbol set selected by the precision-resolution symbol-selection algorithm, and let $\mathcal{I}'$ be some other symbol set which respects p, α, $L(t)$, and $R(t)$. Let S and S' be the resulting constrained systems from $\mathcal{I}$ and $\mathcal{I}'$ respectively. If $R(t)$ is monotone non-decreasing, then $\mathsf{cap}(S) \geqslant \mathsf{cap}(S')$.*

Proof: We denote $\mathcal{I} = \{t_1, t_2, \dots\}$ and $\mathcal{I}' = \{t'_1, t'_2, \dots\}$, where we sort the symbols in increasing order, that is, if $i < j$ then $t_i < t_j$ and $t'_i < t'_j$. Let n be the first index for which $t_n \neq t'_n$, i.e., $t_i = t'_i$ for all $1 \leqslant i \leqslant n-1$. By the definition of the precision-resolution symbol-selection algorithm, necessarily $t_n < t'_n$.

Let us now define $\mathcal{I}'' = \mathcal{I} \setminus \{t'_n\} \cup \{t_n\}$, that is, we take $\mathcal{I}'$ and replace t'_n with t_n. We contend that $\mathcal{I}''$ also respects p, α, $L(t)$, and $R(t)$. Since $t_n \in \mathcal{I}$, then we certainly have no problem with the precision p and the minimum resolution α. Also, $(L(t_n), R(t_n))$ has empty intersection with $(L(t_i), R(t_i))$ for all $1 \leqslant i \leqslant n-1$. So our only concern is that $(L(t_n), R(t_n))$ intersects $(L(t'_{n+1}), R(t'_{n+1}))$. But since $t_n \leqslant t'_n$ we have $R(t_n) \leqslant R(t'_n) \leqslant L(t'_{n+1})$ where the first inequality holds because $R(t)$ is monotone non-decreasing, and the second holds because $\mathcal{I}'$ respects $L(t)$ and $R(t)$.

Repeating this process we arrive at the conclusion that $t_i \leqslant t'_i$ for all $i \geqslant n$. Thus, whenever we inspect the set of τ-headers, $|S(\tau)| \geqslant |S'(\tau)|$ and so $\mathsf{cap}(S) \geqslant \mathsf{cap}(S')$. ■

The precision-resolution system described in the previous sections may lead the unwary reader to assume the receiver requires readings which are at least as precise as the precision at the encoder, or conversely, that there is nothing to be gained by having an encoder more precise than the precision of the readings at the receiver side. At this point we can demonstrate this is not true with the following simple example.

Example 17. *Suppose the receiver has minimal resolution $\alpha = 1$, it suffers from a multiplicative error in its measurements,* and *all its readings are rounded up to the nearest integer. That is, a transmitted interval t may be read as any integer in $[\lceil(1-\delta)t\rceil, \lceil(1+\delta)t\rceil]$. This would correspond to resolution-restriction functions $L(t) = \lceil(1-\delta)t\rceil - 0.5$ and $R(t) = \lceil(1+\delta)t\rceil + 0.5$, with $\delta = 0.1$. The ± 0.5 is introduced just to be consistent with the fact that the readings are in the* open *interval $(L(t), R(t))$.*

An encoder with precision $p = 1$, the same precision as the readings on the receiver side, has a symbol set consisting of $\{1, 3, 5, 7, 9, 12, 16, \dots\}$. This is because a transmitted 1 may be received as any reading between 0.9 and 1.1, and after the rounding up, it is either 1 or 2. We thus contain it in the open interval $(L(1), R(1)) = (0.5, 2.5)$. Similarly, for example, the reading of a transmitted 3 is contained in $(2.5, 4.5)$, and that of a transmitted 12 in $(10.5, 14.5)$. The allowed intervals are chosen greedily according to the precision-resolution symbol-selection algorithm.

On the other hand, let us consider an encoder with a higher precision of $p = 10$. By using the algorithm, such an encoder has a symbol set $\{1, 2.3, 3.4, 4.5, 5.6, 7.8, 10.1, \dots\}$ which obviously results in a higher capacity.

V. Results Summary

We started by exploring the $(1, \alpha, \theta)$-PR constraint which is a natural extension to (d, k)-RLL. The extension allows more run-lengths to be written than (d, k)-RLL admits, subject to the restriction that the decoder, which suffers from a bounded clock drift, can get no ambiguous reading. In light of this, the RLL restriction to successive run-lengths seems arbitrary.

We further generalized our setting by defining the (p, α, θ)-PR framework, and the (p, α, θ)-PR constraint which realizes the framework. We then showed that as $p \to \infty$, the capacity of (p, α, θ)-PR approaches that of (∞, α, θ)-PR from below.

We concluded by showing the underlying greedy algorithm used to choose the symbols given the parameters of the system. This algorithm may be used in a wide range of parameters which under a weak assumption, ensures that the precision-resolution constrained system achieves the optimal capacity. Since the RLL constraint is mainly used today in conjunction with PRML, it is tempting to merge the PR constraint with PRML. Future work may consider the interplay between the two by considering PRML parameters, e.g., sampling rate, as a set of resolution constraints, perhaps achieving another increase in capacity in some applications.

Though the RLL constraint also fits the framework's requirements, the (p, α, θ)-PR constraint seems more natural, and it achieves a higher capacity which does not vanish for $p > 1$ as in the case of RLL. This is especially appealing for engineering reasons: having only an RLL solution, any improvement in precision, e.g., faster clocks, requires a lower clock-drift or else the capacity may drop to zero. However, in the (p, α, θ)-PR constraint, the two parameters of precision and resolution factor, are independent. Thus, they allow improvement of one without the other, always resulting in improved capacity, and offer a trade-off if our goal is to reach a certain prescribed capacity.

References

[1] R. L. Adler, D. Coppersmith, and M. Hassner, "Algorithms for sliding block codes – an application of symbolic dynamics to information theory," *IEEE Trans. on Inform. Theory*, vol. 29, pp. 5–22, 1983.

[2] P. Funk, "Run-length-limited codes with multiple spacing," *IEEE Trans. on Magnetics*, vol. MAG-18, no. 2, pp. 772–775, Mar. 1982.

[3] C. D. Heegard, B. H. Marcus, and P. H. Siegel, "Variable length state splitting with applications to average runlength-constrained (ARC) codes," *IEEE Trans. on Inform. Theory*, vol. 37, pp. 759–777, 1991.

[4] W. H. Kautz, "Fibonacci codes for synchronization control," *IEEE Trans. on Inform. Theory*, vol. IT-11, pp. 284–292, Apr. 1965.

[5] D. Lind and B. H. Marcus, *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, 1985.

[6] B. H. Marcus, P. H. Siegel, and J. K. Wolf, "Finite-state modulation codes for data storage," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 5–37, Jan. 1992.

[7] B. H. Marcus, R. M. Roth, and P. H. Siegel, *Constrained systems and coding for recording channels*. V. S. Pless and W. C. Huffman (Editors), Elsevier, Amsterdam, 1998.

[8] S. Mukhtar and J. Bruck, "Interval modulation coding," Paradise Laboratory, California Institute of Technology, Tech. Rep. ETR040, Oct. 2001. [Online]. Available: http://www.paradise.caltech.edu/papers/etr040.pdf

[9] ——, "Interval modulation coding," in *Proceedings of the 2002 IEEE International Symposium on Information Theory, ISIT2002, Lausanne, Switzerland*, June 2002, p. 327.

[10] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, July 1948.

[11] D. T. Tang and L. R. Bahl, "Block codes for a class of constrained noiseless channels," *Inform. and Control*, vol. 17, pp. 436–461, Dec. 1970.