SPIE Conference on *Advanced Optical Memories and Interfaces to Computer Systems* July 1998 Paper **3468–10**
Proceedings of the SPIE, Vol. **3468,** pp. 64-75 (1998).

Optimizing the holographic digital data storage channel

Geoffrey W. Burr, Jonathan Ashley, Brian Marcus,
C. Michael Jefferson, John A. Hoffnagle, and Hans Coufal

IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120

1. ABSTRACT

Holographic storage has the potential to become a digital data storage technology with fast readout and high density. Computer users have come to expect, however, that data retrieved from their storage devices will be retrieved error-free (with a probability of error $<10^{-12}$). In both conventional storage devices and holographic data storage, achieving this degree of reliability involves a good understanding of the data channel and a combination of careful hardware engineering, signal processing, and coding.

At the IBM Almaden Research Center, we have leveraged the expertise acquired with 1-dimensional, time-dependent data channels found in magnetic and optical data storage systems, to develop unique and highly effective signal processing and coding algorithms to optimize the performance of the 2-dimensional, space-dependent digital holographic data storage channel. Crucial to our efforts has been the high-performance holographic data storage platform we built in 1996. This tool has allowed us to characterize and perturb a real holographic data channel, and implement and evaluate new data-coding and signal processing algorithms. This rapid feedback loop between ideas, implementation, and results both aids in selecting fruitful approaches and yields deeper understanding of the underlying data channel.

In this paper, we discuss the holographic digital data storage channel as divided into five parts: the optical path, pre-processing (how the data gets into the holograms), post-processing (manipulation of raw data just after optical detection), conversion into binary 0's and 1's, and error-correction (using added redundancy). Optimizing the channel involves maximizing the system performance (density, speed) while minimizing complexity (and thus cost) and maintaining the required degree of reliability.

2. INTRODUCTION

By accessing the third dimension of storage media, volume holographic data storage can provide both high density and fast readout [1–3]. Thousands of holograms, each containing a page of data, are multiplexed into a common volume and accessed independently by Bragg–matched diffraction. As the size of this common volume is decreased, density (either volumetric or areal) increases. Since each data page can contain as many as a million pixels [4], reading one thousand pages a second results in a data rate of 1 Gbit/second.

As with any real-world data transmission or storage system, a holographic storage system is a noisy data channel. System optimization is a matter of getting data from input to output at the desired user bit-error-rate (user-BER), while maximizing the desirable properties of the system (density, capacity, and speed) and minimizing the undesirable properties (cost and complexity). In this paper, we describe some of the insights into this optimization process obtained from the 'DEMON' platform [5] at IBM Almaden.

3. THE HOLOGRAPHIC DIGITAL DATA CHANNEL

In a holographic storage system, data passes through a simple 5-step cycle between the time it is passed to the system as input, and the later point at which it is recalled by the user. As illustrated in Figure 1, these steps can be grouped into Pre-processing, the optical path, Post–processing of the retrieved 'analog' data, binarization back into digital data, and error–correction. The following sections describe these steps in more detail; to introduce the physical components involved, we begin with the optical path.

[0]To contact G. W. Burr, Email: *burr@almaden.ibm.com*; Tel: (408) 927–1512; Fax: (408) 927–2100.

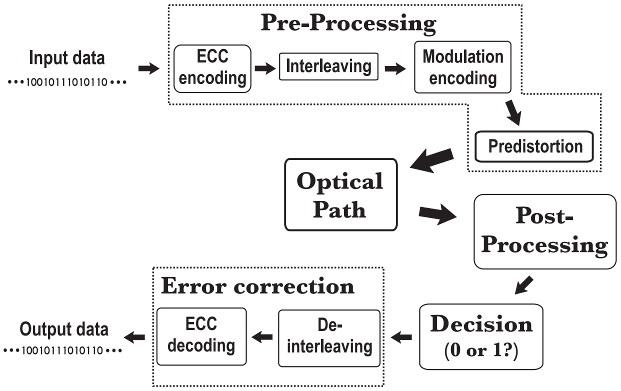

Figure 1: Holographic storage: cycle of data.

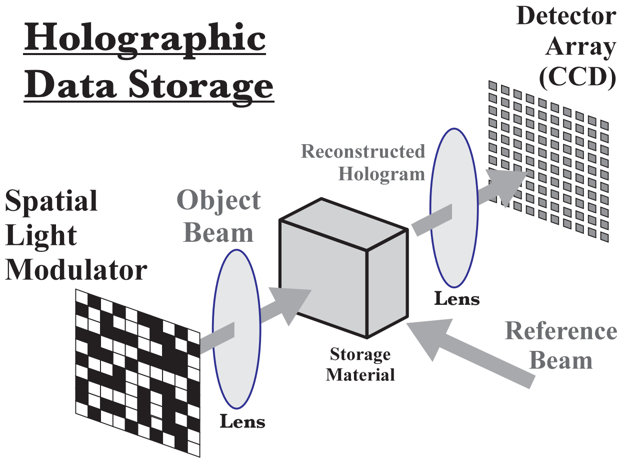

Figure 2: Components of a holographic storage system.

3.1 The optical path

Figure 2 shows the basic components of a digital holographic storage system. A block of photosensitive storage material is surrounded by pixellated input and output devices. To record data, one laser beam passes through the spatial light modulator (SLM) to pick up the input information (the 'object' beam), and meets a second coherent 'reference' beam in the material. A hologram is recorded in the index of refraction of the media. Re-illuminating the hologram with the original reference reconstructs a weak copy of the original information–bearing beam, which is then imaged onto a pixellated detector array. When each of the SLM pixels is accurately imaged to a detector pixel, the bright-or-dark state of a pixel during recording can be successfully detected at some later time during readout—hence, digital data has been stored and retrieved. If the hologram is stored in a thick material, the reconstruction will disappear when the readout beam is changed slightly in incident angle or wavelength. This new reference beam can then be used to store an independently accessible page of data. This has been used to store as many as 10,000 pages in the same $\sim 1cm^3$ block of material [6].

The basic noise trade-off in volume holography is between the finite dynamic range of the recording material and the fixed noise floor of the system. For instance, the electronic detection process at the camera tends to contribute the same amount of noise no matter how bright the hologram. However, as the number of holograms superimposed in the same volume (within the same 'stack' of holograms) increases, the amount of power diffracted into each hologram reconstruction and the resulting signal–to–noise ratio (SNR) decreases. The same problem tends to limit readout rate as well.

Even if all other noise sources are negligible, then there will be a certain hologram strength at which the SNR is inadequate for error–free detection. The number of detected electrons per pixel can be written as

$$n_{electrons} \propto M/\#^2 \, P_{readout} \, \frac{t_{readout}}{M^2 \, N_{pixels}}, \tag{1}$$

where M is the number of multiplexed holograms, N_{pixels} the number of pixels per hologram, $t_{readout}$ the integration time of the camera, $P_{readout}$ the power in the readout beam, and $M/\#$ is a material/system constant [7]. The storage capacity is MN_{pixel} and the readout rate is $N_{pixel}/t_{readout}$. (Storage density is MN_{pixel} divided by the volume or area of each hologram 'stack'.) An increase in either capacity or readout rate leads to a decrease in the number of signal electrons [8]. As this signal strength approaches the number of noise electrons, the BER of the system will rise and the fidelity of the storage system will not meet the promised specifications.

While the constant noise floor is usually of primary importance, any additional noise sources will use up part of the SNR budget. As a result, the minimum acceptable number of signal electrons gets larger, and the capacity of the system is reduced. Noise sources in holographic storage include the following:

- *Change in the readout conditions.* This can occur, for instance, when the recording alters the properties of the recording material, causing unwanted changes in the reference beam path between the time the hologram is recorded and the time it is reconstructed [9–11]. Often, the reference beam angle or wavelength can be tuned to optimize the diffraction efficiency and partially compensate for this effect [9].

- *The detector array doesn't line up with the array of pixels in the reconstructed hologram.* This includes errors in camera registration, rotation, focus, tilt and the magnification of the image.

- *The detector is receiving undesired light,* either from light scattering off the storage material, crosstalk from other stored holograms (inter–page crosstalk [12]), or crosstalk between neighboring pixels of the same hologram (inter–pixel crosstalk [13,14]). Note that while crosstalk contributions scale with the strength of the holograms, the scattering depends only on readout power and the optical quality of the components. Inter–page crosstalk tends to build up as many closely–spaced reference beams are used within the same stack. Inter-pixel crosstalk is essentially diffraction–induced low–pass filtering of the pixellated data page. The system then has a broad point-spread-function, and the sharply-defined input SLM pixels become blurred at the output detector array. This occurs when an aperture is introduced to increase density by reducing the size of each stack within the material.

- *There are brightness variations across the detected image.* This can be a problem if a single threshold is used across the image to separate the pixels into bright and dark and assign binary values. These fluctuations can be caused by the SLM, the optical imaging, or the collimation and beam quality of the laser beams themselves. Such variations tend to be deterministic—they don't vary from hologram to hologram.

Given these many noise sources and the need to read back holograms and make bright–vs–dark distinctions with high fidelity, how can one maximize the desirable qualities of the system such as capacity and readout rate? Here are some options:

1. From Equation 1, we can increase capacity or readout rate by increasing $P_{readout}$ (buying a bigger laser) or by increasing $M/\#$ (getting a better storage material) [15].

2. Pre-process at the spatial light modulator to either increase signal values [16] or reduce the deterministic variations which are reducing the SNR [17].

3. Post-process at the detector array in order to remove a known point-spread-function, with varying degrees of feedback or sequence estimation [13, 18–20]. (Although deterministic variations can also be smoothed out with post–processing, this is best done during pre–processing. In essence, pre–processing knows for sure which pixels are ON and OFF; post–processing doesn't.)

4. Use a low–pass modulation code which avoids pixel combinations which are prone to inter–pixel crosstalk [14,21].

5. Use a decision scheme which produces fewer errors from the same SNR, either with adaptive thresholding [22], or by encoding at the SLM with a balanced modulation code [5].

6. Use interleaving [23] and strong error–correction [24] to produce the same target user-BER from a more error–prone stream of raw binary data.

7. Optimize the physical dimensions of the input and output pixel arrays and of the aperture at the hologram, in order to maximize the storage density.

8. Arrange the recording exposures so that the BERs of the first- and last-written holograms are equal, reflecting any differences in the noise environment experienced by each.

9. Use more than one 'gray' level per pixel, so that each pixel represents more than one bit of information.

We discuss points 2–6 in the remainder of this section, and take up points 7–9 in the next section.

3.2 Pre-processing

The pre–processing part of the system begins with the input of user data and ends when each pixel in the spatial light modulator is set to the desired bright, dark, or in–between gray value. This includes whatever data encoding steps are desired, for modulation or error–correction codes. We will describe such codes at their decoding stages, and assume that any decoder is accompanied by the corresponding encoder at the appropriate point in Figure 1. Note that there is not necessarily a one-to-one correspondence between each SLM pixel and a user bit, especially when using modulation coding.

Pre–processing also includes manipulations of the amplitude and phase of the pixels of the spatial light modulator. By pre–distorting the amplitude of the SLM pixels, deterministic variations in the detected brightness of the ON pixels at the detector array can be removed [17]. Many of these variations are present in an image transmitted with low power from SLM directly to the detector array. Once the particular pattern of non-uniform brightness levels are obtained, one simply modulates each SLM pixel to compensate, either with the 'gray' level response of the device, or by modulating the individual exposure time of the pixel during the recording of the hologram. At low density, BER improvements of more than 15 orders of magnitude are possible [17]. More importantly, at high density, inter–pixel crosstalk (which is deterministic for a given data page) can be suppressed and BER improved from 10^{-4} to 10^{-12} [17]. Additionally, one can increase the contrast between ON and OFF pixel states provided by the SLM, by using interferometric subtraction while recording the hologram to reduce the amount of light received at the 'OFF' detector pixels [17]. Another use of the predistortion technique is to implement gray-scale holographic data pages [25], which we discuss in Section 4.3.

One can also modify the phase of the SLM pixels and implement partial response pre-coding [16]. Here, pixels are set to modulate two states of opposite phase (that is, plus 1 or minus 1). After the object beam passes through a small aperture, neighboring phase–modulated pixels blur into each other. If the detector array is translated by half of the pixel spacing, then a +1 next to a +1 (or -1 next to -1) at the spatial light modulator will result in constructive interference over the straddling detector signal. Conversely, a +1 next to a -1 will result in destructive interference and a detected OFF pixel. Thus contrast at high density can be improved by using the diffraction–induced blur to one's advantage. Partial response is fairly easy to use when the detector is offset only in one dimension, as opposed to being offset both horizontallly and vertically. This is due to the much more complex pre-coding [26]. In addition, the square aperture creating the point-spread-function must either be rotated 45° or replaced by a circular aperture.

3.3 Post-processing

After a data page passes through the optical channel (having been holographically stored and retrieved), each detector pixel receives an optical signal. The detected photoelectrons are then converted to 'camera counts' by an analog-to-digital converter. These are digital representations (typically 8 bits) of the analog signal values, making mathematical manipulations of the pixel values possible. Even so, we refer to these as the analog pixel values, so that the pixel values are not 'digital' until the decision stage.

Several methods for improving performance by manipulating these analog pixel values have been developed. Typically, these post-processing techniques perform a convolution with a small kernel designed to un-do the known broadening caused by the band-limiting optical channel. Such a kernel can be derived by simply inverting the channel's spatial frequency response (zero-forcing equalization [19]). While this removes the deterministic inter-pixel crosstalk, it can amplify the random noise such as optical scatter or electronic detector noise. In addition, although the inter-pixel interference is occurring coherently (electric field amplitude), the detector pixels (and thus the reported camera counts) can only measure intensity. It appears, though, that this channel non-linearity can be compensated by taking the square root of the camera count values before any manipulation [19]. Wiener filtering attempts to remove inter–pixel crosstalk by building a filter which minimizes the mean–squared error between the filtered output and the intended output [18]. This balances the removal of inter–pixel crosstalk against the amplification of random noise.

3.4 The decision stage

After any post-processing steps, the decision stage converts the received analog values back to binary data. This reduces to deciding whether each pixel is supposed to be OFF or ON. Errors are created when bright OFF or dark ON pixels fool the detection process into assigning them to the wrong category.

One simple method is to use a single, 'global' threshold for the whole page. This generally works poorly because the data pages tend to be fairly non-uniform in brightness: the OFF pixels from the bright center can approach the intensity of the ON pixels in the dim corners. This spatial variation can push pixels across the global threshold, causing errors even in situations where the ON and OFF pixels are well-separated locally.

As a result, it is advantageous to have the threshold vary across the page. This can be done by adapting the threshold using the last few pixels and the decisions made on them [22]. The formula for the local threshold can

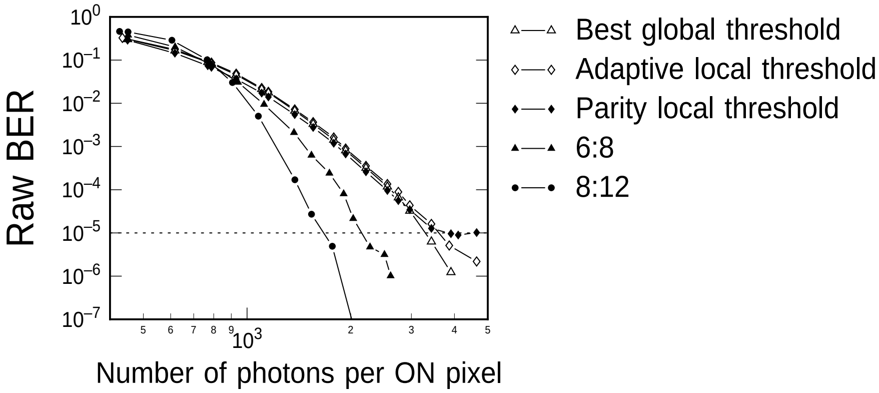

Figure 3: BER performance vs. signal strength in the presence of a fixed noise floor.

be simple, and the overhead (number of pixels not representing user data) can be quite low. This results in a code rate (the fraction of pixels that represent user data) close to unity. The disadvantage, however, is that a few wrong decisions can propagate to a large run of errors [27].

Alternatively, a local threshold can be derived for a small block of pixels by using 'parity' information about the number of ON pixels within the block [27], transmitted to the decoder using a separate portion of the data page with low code–rate but reliable differential encoding (ON–OFF means binary 0, OFF–ON binary 1) [2]. Knowing the tabulated sum of camera counts and the number of ON and OFF pixels which produced that sum, one can obtain a reliable estimate of the local threshold. Since errors made in decoding the parity information are relatively unimportant, the performance of this scheme is dominated by the performance of the local threshold [27]. The code rate of parity thresholding depends on the size of the local block—for a block of 16×16 pixels, the code rate is approximately 0.94.

For stronger performance than these thresholding schemes, one can use a balanced modulation code [5]. Here a small block of pixels is encoded so that a given fraction (for instance, half) of the pixels are guaranteed to be ON. The number of bits that can be conveyed is $\log_2$ of the number of possible combinations. For instance, with 8 pixels there are 70 possible combinations of 4 ON and 4 OFF, which allows one to store 6 bits of information for a code rate of 75%. Decoding is simply a matter of sorting the pixel values and then looking up the appropriate label. This has the advantage of implicitly finding the best local threshold—if there is a threshold which will work, the decoder will find it and use it. Alternatively, a correlation detector can be used instead of sorting [5]. This has a small speed advantage for simple codes, but correlation detection also allows one to implement more complex codes in which the 'Hamming distance' between codewords is larger. For the simple 6:8 code described above, a symbol error occurs (that is, 6 random bits are delivered) whenever the dimmest ON pixel is exceeded by one of the OFF pixels. For a more complex code (one example we have implemented is an 8:12 code with 66% code rate) [5], this condition would still be decoded correctly because of the correlation decoder and the increased Hamming distance. This gives a stronger performance, at the cost of decoding complexity and the lower code rate.

Figure 3 shows the raw-BER performance of four decoding strategies as a function of signal strength (in photons detected per ON pixel) [5]. The noise source is approximately 160 noise photons of electronic detection noise (referenced through the ~30% quantum efficiency of the detector). As expected, the modulation codes provide stronger performance in exchange for their lower code rate. At the same raw–BER, the modulation codes can detect weaker holograms, which implies that more holograms can be stored. The codes win this tradeoff, however, only if the increase in the number of holograms pays for the loss per data page reflected in the lower code rate. Recently, we developed an experimental measurement technique which can quantify this tradeoff, which we describe in Section 4.2.

More complicated decision schemes can be used which essentially combine the post-processing and decision stages. These include maximum likelihood sequence estimation with the Viterbi algorithm [13], Viterbi in combination with decision feedback [13], and parallel detection [18]. The main drawback to such techniques is finding an implementation which maintains most of the performance, but is streamlined in such a way that its electronic implementation can run at the required data rate (1 Gbit/second divided by the number of separate data 'taps' on the detector array).

Before making the experimental measurements in Figure 3, we turned up the gain of the analog-to-digital (A/D) converter so that the incoming signal filled most of the dynamic range. If we had not done this, then the quantization of the detected analog signals might assign ON and OFF pixels of similar brightness—that would otherwise be slightly separated—to the same camera count value, leading to an error. Figure 4 illustrates the effect of this 'quantization noise', showing BER as a function of the quantizing resolution (represented as the $\log_2$ of the number of camera counts separating the means of the ON and OFF pixels). In this simulation, the same two Gaussian noise sources (SNR $\equiv (\mu_1 - \mu_0)/\sqrt{\sigma_1^2 + \sigma_0^2} = 3.5$, $\sigma_1/\sigma_0 = 3$) were applied, only varying the resolution of the quantization. This shows that for the purposes of decoding performance, one needs only $\sim$ 2.5 bits of quantization. In other words, it is not important to know the safety margin by which the ON pixels exceed the threshold. However, for the predistortion technique or any of the post-processing techniques or any decision feedback scheme, we require accurate knowledge of the brightness of pixels. The gain of the A/D converter should then be adjusted so that the brightest ON pixels still fall within the dynamic range of the camera. Maintaining this condition with 8 bits of dynamic range and the brightness distributions listed above leaves approximately 130 camera count levels (7 bits) betweeen the means.

3.5 Error-correction

The modulation codes described above improve performance by adding redundancy—but this redundancy is distributed over the pixels which make up the modulation codeword and provides its advantages at the decision stage. In contrast, error-correction coding (ECC) works on the binary data by adding explicitly redundant bits to each block of incoming stream of user bits. These extra bits are chosen in such a way that, upon decoding of this block, a small number of errors can be detected, located, and corrected. (Essentially, decision errors disrupt the mathematical inter–relationships among the bits.) A widely used family of ECC are Reed-Solomon (R-S) codes, which operate on codewords of length 2^n-1 symbols (where each symbol contains n bits). If $2t$ of these symbols are reserved for redundancy, then the R-S code can correct up to t symbol errors. (This works the same no matter how many bits are in error within the erring symbols, and whether these symbols carry user data or not.)

Such a code pays for its code rate by dramatically lowering the demands on the previous stages, as measured by the raw-BER of the decoded data stream being presented to the ECC decoder. With a relatively small overhead (t=28 redundant symbols, or a code rate of 0.89, to be able to correct up to 14 symbol errors), an 8-bit-per-symbol, 255 symbol R-S code can correct from a permissive raw-BER of 10^{-3} down to the stringent output specification of 10^{-12} user-BER [27]. Without ECC, the raw-BER would have to be 10^{-12}, reducing capacity much more than the modest ECC code rate. Thus judicious use of ECC increases capacity [24]. In order to keep a localized cluster of errors from overwhelming a codeword, interleaving can be included. This insures that the symbols of each codeword come from detector pixels that are distributed across the data page. In addition, if some regions of the data page are known to be more prone to error, a matched interleaver (which uses this knowledge to disperse symbols from danger regions across many codewords) will out-perform an interleaver which assigns symbols to codewords randomly [23]. The interleaver is also a good place to avoid using 'dead' pixels in the SLM or detector array. Often, this costs nothing at all, because the data page carries a total number of bits which is not an integral multiple of the number of bits per ECC codeword. If some pixels have to go to waste, why not those that don't work anyway?

4. CHANNEL OPTIMIZATION

In this section, we describe several methods for optimizing the holographic digital data channel. These range from encoding of pixels to selection of components to choice of recording exposures.

4.1 SLM and detector fill factors

Some component choices have an obvious effect on system performance. For instance, the size of the central diffraction order of the SLM scales as

$$D_N \equiv \frac{\lambda f}{\delta}, \tag{2}$$

where λ is wavelength, f is the focal length of the object beam lens, and δ is the spacing between SLM pixels. Essentially, the image will be at the Nyquist sampling condition if a centered square aperture at the Fourier plane has sides of length D_N. If it were any smaller, spatial frequency components which represent data would be cut off. As a result, areal density can be increased by having a short wavelength, short focal length, and large pixels. The limit here is how far the lens design can be pushed to accept a large input field (N_{pix} pixels times δ) yet still maintain a short focal length.

However, maintaining the Nyquist sampling condition does not necessarily guarantee low BER. The aperture is essentially acting as a spatial low–pass filter, so that the transmitted image is convolved with a point-spread-function (PSF). Since the PSF is the Fourier transform of the aperture, larger apertures mean a narrower PSF and thus less blurring of pixels into their neighbors. Although this reduces the data density per hologram, it leaves more

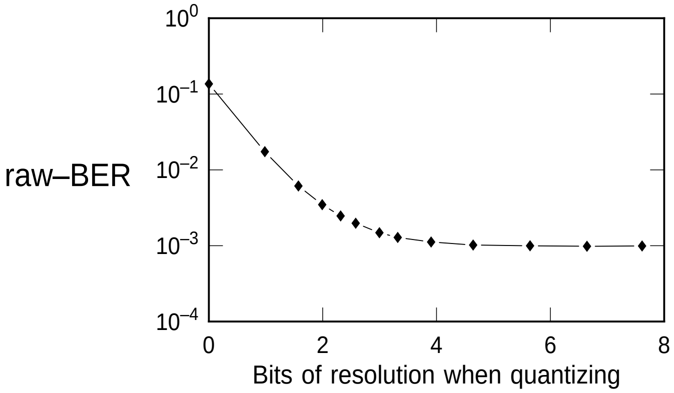

Figure 4: BER vs. quantization resolution, expressed as $\log_2$ of the number of camera count levels between the means of the ON and OFF pixels.

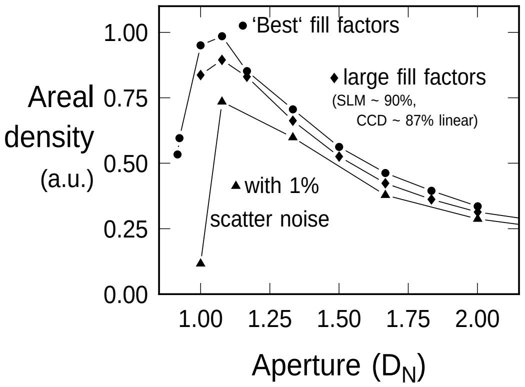

Figure 5: Normalized areal density as a function of aperture size (as a function of the Nyquist–sampling aperture).

of the SNR budget available for combating background noise, which in turn allows one to write more holograms. In addition to this tradeoff, there is the effect of the fill factors of the SLM and detector array. (The fill factor is the fraction of the space assigned to a pixel which is actually used—modulated in the case of the SLM, or used for collecting photoelectrons in the case of the detector pixel). Since the shape of the output pixel is a convolution of the original SLM pixel and the PSF, it might make sense to have a small SLM fill factor, to reduce the inter–pixel crosstalk yet still allow a small aperture. (An additional effect of small SLM fill factor is to dramatically reduce the power efficiency of the SLM, since much of the illuminating light is dumped on the dead space between active pixel elements). On the detector array, it might make sense to have very small pixels so that the inter–pixel crosstalk can be effectively removed. This is the Nyquist sampling condition—for the particular aperture of sides D_N, the contribution of neighboring pixels goes to zero at the exact center of the pixel. So point-detectors would be able to avoid all inter–pixel crosstalk. Unfortunately, point-detectors don't collect very many photons, so it's impossible to build up enough signal to overcome the fixed background noise. These two opposing trends—small pixels to resist inter-pixel crosstalk, large pixels to collect enough signal—create an optimum fill factor [28]. It turns out that the optimum pair of fill factors changes for different aperture sizes.

We have studied the interplay between electronic detector noise and inter–pixel crosstalk using a simulation which includes the effects of component fill factors and aperture size [28]. The achievable density is shown graphically by the top curve of Figure 5, where normalized areal density is plotted as a function of the aperture size (itself normalized to the Nyquist aperture). The assumption here is simple global thresholding and a raw-BER target of 10^{-4}. We have found that the need to collect photons is more important than suppressing interpixel crosstalk, however, which means that using relatively large fill factors (linear SLM f.f. of 90%, linear CCD f.f. of 87%) costs little in terms of density (middle curve of Figure 5). We have also looked at the case where scatter noise is added to the detector noise and inter-pixel crosstalk [29]. This optical noise contribution tends to decrease as the fill factor of the detector pixel decreases, pushing the optimum pixel size lower (although large fill factors still work pretty well). As one would expect, adding another noise source tends to reduce the portion of the SNR budget left for writing holograms (that is, for allowing the signal levels to drop), and the achievable areal density reflects this (bottom curve of Figure 5). What is most surprising is that when the scatter noise power exceeds $\sim$1% of the received signal power, then the raw-BER target of 10^{-4} is unachievable (all of the SNR budget is used up by scatter noise and inter–pixel crosstalk). This is a common situation in these simulations: as a noise source increases, the capacity or density at a given BER target decreases slowly at first, but then rapidly drops to zero. At this point, use of modulation coding or the pre-distortion technique is doing more than just increasing capacity—it makes it possible for the system to store and retrieve holograms at all under these conditions.

4.2 Flat-BER recording schedule

The recording schedule was originally developed to allow multiple holograms to be recorded to the same diffraction efficiency, η, in read–write holographic materials such as photorefractive crystals. Since the first holograms written must last through the erasure caused by writing later holograms, the diffraction efficiency is equalized by carefully

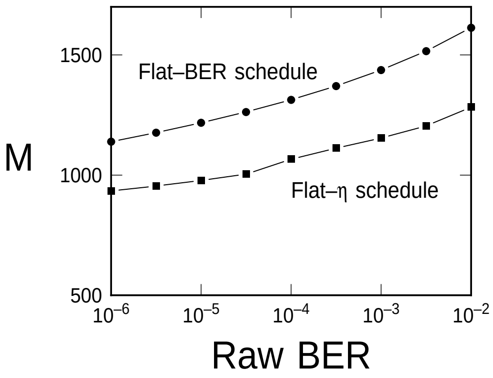

Figure 6: Number of holograms that can be stored as a function of raw-BER. By tolerating a higher raw-BER, more holograms can be superimposed. Similarly, capacity can be improved by arranging recording exposures to equalize BER instead of diffraction efficiency [27].

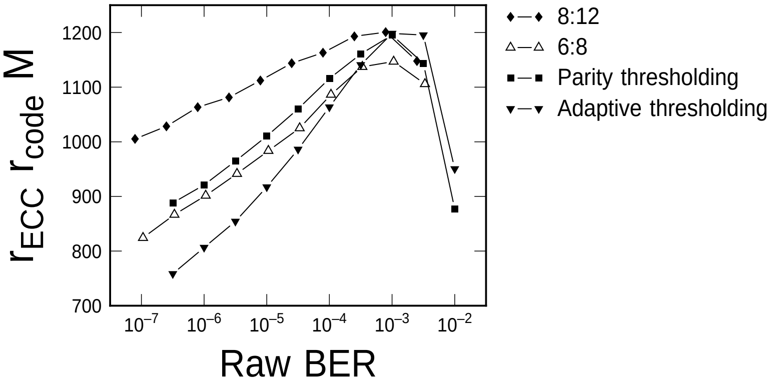

Figure 7: Total capacity in user bits as a function of the system raw-BER. The final user-BER is always 10^{-12}, and two modulation codes are compared against two local thresholding techniques. The strong raw-BER performance of the codes is counter-balanced by their lower code rate. The presence of the ECC coding makes it possible to find an optimal raw-BER [27].

scheduling the recording exposures. Initial holograms are written for a longer exposure time and thus grow to a large diffraction efficiency, which is then exactly compensated by the erasure due to the remaining exposures—and the first and last holograms end up with the same diffraction efficiency.

Recently, we developed a variation on this recording schedule, which uses the same mathematical algorithm, but attempts to equalize the **raw-BER** of the first and last holograms. This has the effect of increasing capacity, because the last hologram is typically not experiencing the same amount of noise as the first hologram. Recording to the same diffraction efficiency is overkill for the last hologram, because this hologram ends up with a raw-BER which is better than required. This extra dynamic range can be used to record additional holograms, creating an increase in total capacity. This is shown in Figure 6, where the number of holograms that can be stored is plotted against raw-BER, for both a conventional, flat–η schedule and the new flat–BER schedule.

The main motivation for this new schedule, however, was not the increase in capacity shown by the measurement in Figure 6, but just the capability of making such a measurement at all. To measure the number of holograms as a function of raw-BER would otherwise be an exhausting series of multiple hologram exposure experiments. Instead, a simple set of repeated measurements of the same hologram gives the relation between exposure time and raw-BER. Using the mathematics of the flat–BER recording schedule, this results in a unique value of M—the number of holograms that can be stored—for each target raw–BER.

We have used this procedure to measure and optimize the capacity, in user bits, of our ‘DEMON’ holographic storage system operating at low density (large Fourier transform aperture) [27]. With the M vs raw-BER data made possible by the capacity measurement described above, we combine the code rate of the modulation coding and the dependence of the ECC code rate on raw-BER. The result of such a study is shown in Figure 7, where two modulation codes (6:8 and 8:12) are compared against two local thresholding techniques on the basis of total user capacity. Note that the tendency, as raw-BER gets larger, for the number of holograms to increase and the ECC code rate to decrease creates an optimal raw-BER operating point for the system ($\sim 10^{-3}$ in this case).

4.3 Gray-scale

We have also used the capacity estimation technique to study the capacity provided by gray–scale data pages, in which each pixel takes one of g briqhtness levels. In the absence of noise, each pixel is then capable of transmitting $\log_2 g$ bits of data. However, the finite SNR budget has been divided into $g-1$ parts. The tradeoff between more bits per pixel and less SNR with which to multiplex holograms results in a optimum number of gray levels. If only background noise is present, then five gray levels would be the best choice. If, however, both background noise and signal–dependent noise sources are present, then the optimum number of gray levels and the resulting capacity gain decreases [25].

We used the predistortion technique [17] to create holographic data pages with multiple gray levels. A histogram

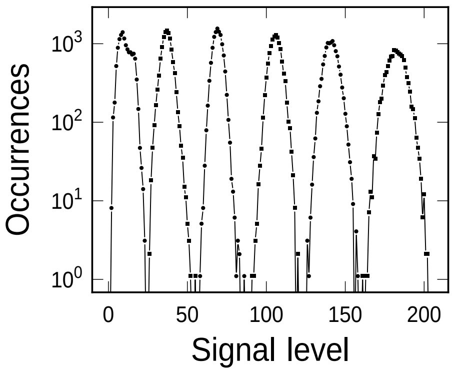

Figure 8: Histogram of received values for grayscale hologram with six levels ($\log_2 6 \sim 2.58$ bits per pixel).

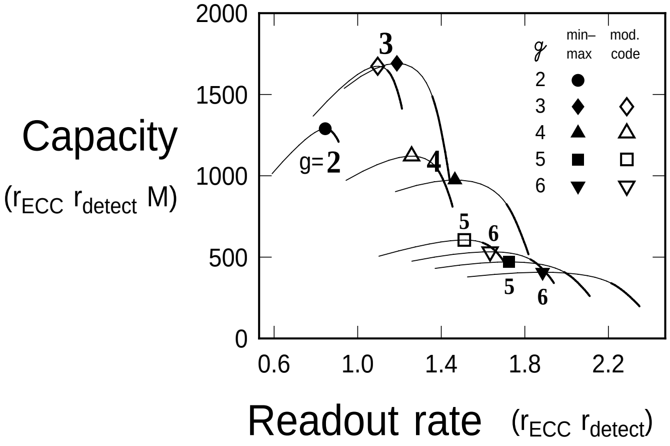

Figure 9: Capacity vs. readout rate [25], measured with experimental capacity estimation technique [27].

from a six level hologram is shown in Figure 8. We developed several gray-scale modulation codes and one local–thresholding technique (which derived the $g-1$ thresholds for each block from the minimum and maximum detected signals within the block). The modulation codes, described in Reference [25], are convenient because they explicitly encode from and decode back to binary data, avoiding the question of how to efficiently take binary data and turn it into, say, five gray levels per pixel. The capacity performance of these codes and gray levels is shown in Figure 9, where user capacity in bits is plotted against readout rate [25]. Each symbol indicates the maximum capacity point, while the line indicates the capacity–readout rate tradeoff as the ECC coding is varied from weak coding (right side, t=4 bytes of an 8-bit-per symbol, 255 symbol R-S code) to strong coding (left side, t=47). Since distance along this curve is not linear in t, we indicate the point t=16 by changing the curve from solid to dotted. Roughly speaking, the solid line indicates ECC solutions that are available off–the–shelf; for operating points chosen along the dotted line, the complexity of the ECC decoder may reduce readout speed.

4.5 Variations in hologram strength

We separated the noise sources described earlier into two types: those which are truly random noise (from the detection electronics or optical scatter) and the deterministic variations in signal strength (fixed pattern noise or inter-pixel crosstalk). These sources all tend to make pixel values deviate from the average ON (or OFF) value for the page. One question worth considering is what happens when the mean values themselves are varying from hologram to hologram (because the diffraction efficiency is varying). The dependence of BER on signal power at low density is shown in Figure 3. If we could count on the diffraction efficiency of all the holograms to be equal, we would multiplex holograms until the page-wide raw-BER reached the given target value (say, raw-BER of 10^{-3} with ECC coding designed to bring that down to the user-BER target of 10^{-12}). In reality, however, the diffraction efficiency of holograms varies from page to page. So every once in a while there is an weaker-than-average page which is decoded at a raw-BER of 10^{-2}, resulting in a user-BER above the 10^{-12} target, and raising the average user-BER. To combat this, we need to increase the average mean of the pages and thus decrease the raw-BER target that we try to hit with this average-strength page. Here we try to estimate how much the average page strength has to increase, and show the resulting cost in user capacity.

We use the dependence of raw–BER vs. signal strength for the 6:8 code as shown in Figure 3, and assume that a 8-bit-per-symbol, 255-byte-long Reed–Solomon code is being used. With $t = 14$ bytes of error–correction, this can take a raw–BER of 10^{-3} down to a corrected user-BER of 10^{-12}. We assume that the user-BER roughly follows the raw-BER raised to the 14^{th} power (times a correction factor which puts the curve through 10^{-12} user-BER for 10^{-3} raw-BER). This implies that a change of raw-BER by one order of magnitude causes the user-BER to swing by 14 orders of magnitude. The combination of the 6:8 curve from Figure 3 (signal strength → raw-BER) and this 1:14 dependence (raw-BER to user-BER) results in Figure 10. This plot of user-BER against signal strength shows that a small decrease in diffraction efficiency below the average diffraction efficiency results in a huge jump in user-BER. Assuming that the signal strength follows some probability distribution $p_{signal}(\eta)$, then the aggregate user-BER will

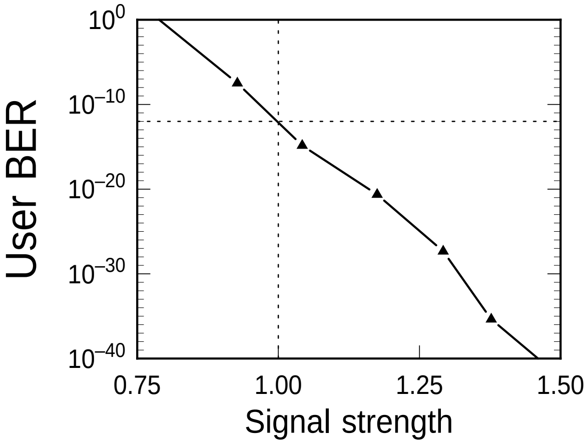

Figure 10: Rough dependence of user-BER on input signal strength, relative to the design point of correcting a raw-BER of 10^{-3} down to 10^{-12}. As a result, small variations in the diffraction efficiency of the holograms result in large swings in user-BER.

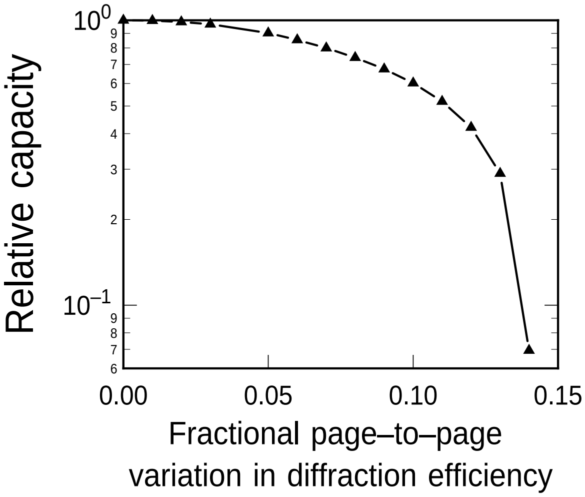

Figure 11: Loss of density resulting from variation of the diffraction efficiency of the holograms. Pages which are weaker than the designed–for average–strength page have a much higher user-BER. The required increase in the strength of the average page results in a loss of density (because not as many holograms can be superimposed).

be

$$\text{Average user BER } = \int p_{signal}(\eta) \quad \text{user BER}(\eta) \;\; d\eta. \tag{3}$$

As the variation in diffraction efficiency of the pages increases, the average user-BER starts to exceed the previously designed–for target because of the weak pages. (If we got to average the log of the user-BER, then everything would balance out.) Note that to calculate the average user–BER, we take the user-BER(η) to be 0.5 for holograms with relative signal strength less than ∼0.8 (see Figure 10). To counteract the increase in average user-BER, we have to increase the average diffraction efficiency and correspondingly reduce the number of holograms that can be stored. (Since diffraction efficiency scales as one over the number of holograms squared, having to increase average diffraction efficiency by a factor f reduces the number of stored pages by $\sqrt{f}$.) This reduction in capacity is shown in Figure 11, where the horizontal axis shows the ratio of the standard deviation to the mean of the Gaussian distribution of diffraction efficiencies. Note that after a certain point, the user-BER and raw-BER for the average page no longer affect the design point—the errors in user data come solely from the weakest data pages. The performance of the system is then increased by raising the average diffraction efficiency just to strengthen the weakest pages.

One point to consider, however, is that we locked in the particular choice of ECC before including the effects of varying page strength. We assumed that the ECC code should correct down to exactly the target user–BER, and that the average diffraction efficiency should then be increased to account for the signal variation among pages. An alternative would be to pay additional ECC code rate to correct farther than 10^{-12}, which might then mean that signal strength might not have to increase at all in order to meet the 10^{-12} target. However, what seems to cause all of the trouble (especially for large swings in signal strength) is the steep slope of the user-BER vs. raw-BER dependence. This slope could be decreased by choosing a **weaker** ECC code. And, as a final complication, it may be that, if the variations are primarily introduced during recording, then the best systems solution is to check each hologram after recording and re-record those pages which are the extremely weak outliers.

5. CONCLUSIONS

We have described the data channel found in digital holographic data storage systems. These include the optical data path (consisting of many noisy parallel channels between SLM and detector pixels), preprocessing (manipulation of the states of the SLM pixels), post-processing (manipulation of the 'analog' detected signals in order to remove inter–pixel crosstalk), the decision stage (where pixels are classified resulting in binary data), and the error–correction stage (where redundant bits are used to locate and correct bit errors). We have described various choices available in these stages, as well as additional steps that can be taken to optimize the digital holographic data channel. These include careful choice of components such as the Fourier transform aperture and device fill factors (both SLM and detector), gray-scale encoding (more than 1 bit per pixel), and scheduling of the recording exposures to equalize the raw-BER of the holograms. Additionally, we showed that the detection process is relatively robust when the analog signal values are quantized with low–resolution, and that page-to-page variations in the diffraction efficiency of holograms decreases capacity.

6. ACKNOWLEDGEMENTS

The authors gratefully acknowledge helpful discussions with members of the Holographic Data Storage Systems consortium. This work was partially supported by the U. S. Defense Advanced Research Projects Agency through the Holographic Data Storage Systems consortium under agreement MDA972-95-0004.

7. REFERENCES

[1] D. Psaltis and F. Mok. Holographic memories. *Scientific American*, 273(5):70, 1995.

[2] J. F. Heanue, M. C. Bashaw, and L. Hesselink. Volume holographic storage and retrieval of digital data. *Science*, 265:749, 1994.

[3] J. H. Hong, I. McMichael, T. Y. Chang, W. Christian, and E. G. Paek. Volume holographic memory systems: techniques and architectures. *Optical Engineering*, 34:2193–2203, 1995.

[4] R. M. Shelby, J. A. Hoffnagle, G. W. Burr, C. M. Jefferson, M.-P. Bernal, H. Coufal, R. K. Grygier, H. Günther, R. M. Macfarlane, and G. T. Sincerbox. Pixel–matched holographic data storage with megabit pages. *Optics Letters*, 22(19):1509–1511, 1997.

[5] G. W. Burr, J. Ashley, H. Coufal, R. K. Grygier, J. A. Hoffnagle, C. M. Jefferson, and B. Marcus. Modulation coding for pixel–matched holographic data storage. *Optics Letters*, 22(9):639–641, 1997.

[6] G. W. Burr, F. H. Mok, and D. Psaltis. Storage of 10,000 holograms in $LiNbO_3$:Fe. In *CLEO 1994*, page 9, 1994. paper CMB7.

[7] F. H. Mok, G. W. Burr, and D. Psaltis. System metric for holographic memory systems. *Optics Letters*, 21(12):896–898, 1996.

[8] K. Blotekjaer. Limitations on holographic storage capacity of photochromic and photorefractive media. *Applied Optics*, 18:57–67, 1979.

[9] R. DeVre, J. F. Heanue, K. Gürkan, and L. Hesselink. Transfer functions based on Bragg detuning effects for image–bearing holograms recorded in photorefractive crystals. *Journal of the Optical Society of America* **A**, 13(7):1331–1344, 1996.

[10] S. Campbell, S.-H. Lin, X. Yi, and P. Yeh. Absorption effects in photorefractive volume–holographic memory systems. i. beam depletion. *Journal of the Optical Society of America* **B**, 13(10):2209–2217, 1996.

[11] S. Campbell, S.-H. Lin, X. Yi, and P. Yeh. Absorption effects in photorefractive volume–holographic memory systems. ii. material heating. *Journal of the Optical Society of America* **B**, 13(10):2218–2228, 1996.

[12] C. Gu, J. Hong, I. McMichael, R. Saxena, and F. Mok. Cross–talk–limited storage capacity of volume holographic memory. *Journal of the Optical Society of America* **A**, 9(11):1–6, 1993.

[13] J. Heanue, K. Gurkan, and L. Hesselink. Signal detection for page–access optical memories with intersymbol interference. *Applied Optics*, 35:2431–2438, 1996.

[14] J. Hong, I. McMichael, and J. Ma. Influence of phase masks on cross-talk in holographic memory. *Optics Letters*, 21:1694–1696, 1996.

[15] G. W. Burr and D. Psaltis. Optimization of the oxidation state of $LiNbO_3$ for large scale holographic storage. *Optics Letters*, 21(12):893–895, 1996.

[16] B. H. Olson and S. C. Esener. Partial response precoding for parallel–readout optical memories. *Optics Letters*, 19(9):661–663, 1994.

[17] G. W. Burr, H. Coufal, R. K. Grygier, J. A. Hoffnagle, and C. M. Jefferson. Noise reduction of page–oriented data storage by inverse filtering during recording. *Optics Letters*, 23(4):289–291, 1998.

[18] M. A. Neifeld, K. Chugg, and B. King. Parallel data detection in page–oriented optical memory. *Optics Letters*, 21:1481–1483, 1996.

[19] V. Vadde and B. V. K. Vijaya Kumar. Channel estimation and intra–page equalization for digital volume holographic data storage. In *Optical Data Storage 1997*, pages 250–255, 1997.

[20] B. King and M. A. Neifeld. Parallel detection algorithm for page–oriented optical memories. *Applied Optics*, 37(26):6275—6298, 1998.

[21] J. Ashley and B. Marcus. Two–dimensional lowpass filtering codes for holographic storage. *IEEE Transactions on Communications*, 46:724–727, 1998.

[22] X. A. Shen, A.-D. Nguyen, J. W. Perry, D. L. Huestis, and R. Kachru. Time–domain holographic digital memory. *Science*, 278:96–100, 1997.

[23] W.-C. Chou and M. A. Neifeld. Interleaving and error correction in volume holographic memory systems. *Applied Optics*, 37(29):6951–6968, 1998.

[24] M. A. Neifeld and M. McDonald. Error correction for increasing the usable capacity of photorefractive memories. *Optics Letters*, 19:1483–1485, 1994.

[25] G. W. Burr, G. Barking, H. Coufal, J. A. Hoffnagle, C. M. Jefferson, and M. A. Neifeld. Gray–scale data pages for digital holographic data storage. *Optics Letters*, 23:1218–1220, 1998.

[26] B. H. Olson and S. C. Esener. One– and two–dimensional parallel partial response for parallel readout optical memories. In *Proceedings of 1995 IEEE International Symposium on Information Theory*, page 141, 1994.

[27] G. W. Burr, W.-C. Chou, M. A. Neifeld, H. Coufal, J. A. Hoffnagle, and C. M. Jefferson. Experimental evaluation of user capacity in holographic data storage systems. *Applied Optics*, 37:5431–5443, 1998.

[28] M.-P. Bernal, G. W. Burr, H. Coufal, and M. Quintanilla. Balancing inter–pixel crosstalk and thermal noise to optimize areal density in holographic storage systems. *Applied Optics*, 37:5377—5385, 1998.

[29] M.-P. Bernal, G. W. Burr, H. Coufal, and M. Quintanilla. Noise in holographic data storage at high areal density. In *CLEO 1998 Technical Digest*, 1998. paper CMF3.